AVA19 (

Reference 2

(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
12 December 2002 (12.12.2002)

PCT

(10) International Publication Number
WO 02/099640 A1

(51) International Patent Classification[7]: G06F 11/00, 1/24

(21) International Application Number: PCT/US01/18324

(22) International Filing Date: 6 June 2001 (06.06.2001)

(25) Filing Language: English

(26) Publication Language: English

(71) Applicant *(for all designated States except US)*: YAHOO INC. [US/US]; 701 First Avenue, Sunnyvale, CA 94089 (US).

(72) Inventors; and

(75) Inventors/Applicants *(for US only)*: MADISON, Justin [US/US]; 1504 Concord, Richardson, TX 75081 (US). RODIGER, Anthony [US/US]; 5004 Lake Vista Drive, The Colony, TX 75056 (US). CHINTALA, Ajay [IN/US]; 16951 Addison Road, Apt. 2108, Addison, TX 75001 (US).

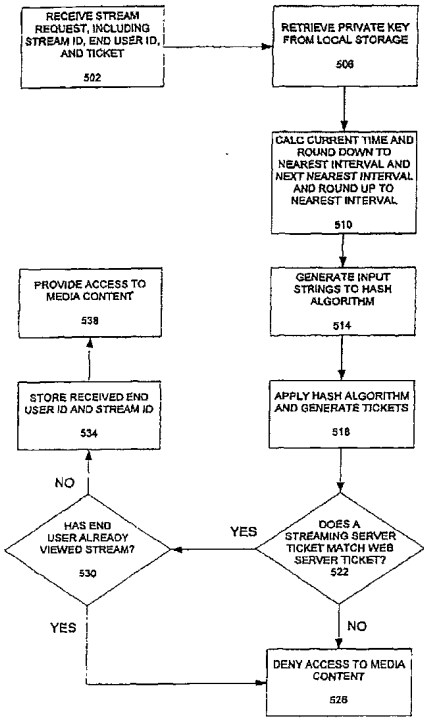
(74) Agents: ROSENTHAL, Lawrence et al.; Stroock & Stroock & Lavan, LLP., 180 Maiden Lane, New York, NY 10038 (US).

(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European

[Continued on next page]

(54) Title: SYSTEM AND METHOD FOR CONTROLLING ACCESS TO DIGITAL CONTENT, INCLUDING STREAMING MEDIA



(57) Abstract: A system and method for controlling access to digital streaming data (502). The media server generates an authorization ticket and compares (522) it to one generated by the web server (518) to determine whether to grant access (530).

WO 02/099640 A1

patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Declarations under Rule 4.17:

— *as to applicant's entitlement to apply for and be granted a patent (Rule 4.17(ii)) for the following designations AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CO, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, UZ, VN, YU, ZA, ZW, ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG)*

— *of inventorship (Rule 4.17(iv)) for US only*

— *of inventorship (Rule 4.17(iv)) for US only*

— *of inventorship (Rule 4.17(iv)) for US only*

Published:

— *with international search report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

SYSTEM AND METHOD FOR CONTROLLING ACCESS TO DIGITAL CONTENT, INCLUDING STREAMING MEDIA

BACKGROUND OF THE INVENTION

1. Field of the Invention

[001] The present invention relates generally to controlling access to digital content and, more particularly, to ticket-based systems and methods for limiting access to streaming media, wherein the ticket is based, in part, on a time component.

2. Description of Related Art

[002] With the advent of the Internet and the World Wide Web, an industry has developed around the delivery of digital content, such as streaming media content. By way of example, streaming media may be used for any of a number of purposes, including entertainment, distance learning and corporate purposes. Entertainment companies stream movies and sporting events, distance learning companies stream educational content, and corporations stream training materials.

[003] With many such uses of streaming media, controlling access to the content is imperative. For example, entertainment companies may charge end-users for each viewing of an item of streaming media, referred to in the entertainment vernacular as "pay-per-view." Similarly, distance learning companies charge students for access to on-line educational courses, and thus for access to streaming media. Corporate content is often confidential, and thus also requires controlled access.

[004] Accordingly, systems have been developed to limit access to streaming media. The current industry standard for limiting access to streaming content involves the streaming media server authenticating end-users before providing the streaming media content. More specifically, the streaming media server typically includes a software plug-in of compiled code that contains the logic for determining whether or not to grant access to the streaming

media. Such an authentication plug-in, however, is often complicated and difficult to develop and maintain. For example, if the need arises to change to logic for granting access to the streaming media content, altering the compiled plug-in on the streaming media server is difficult. Furthermore, with all of the logic residing at the streaming media server, the streaming media server must have direct access to a database or distributed message passing service. Similar problems exist with systems used for controlling access to other forms of digital content. Accordingly, a need exists for an improved system and method for controlling access to digital content, particularly streaming media content, and authorizing end users.

3. Summary of the Invention.

[005] The present inventions solves this and other needs by providing a system and method for controlling access to digital content, such as audio, visual, video, textual and streaming media. One system and method according to the present invention controls access to streaming media and includes a web server, media server and end user processor, such as a personal computer, coupled to a network.

[006] In operation, the web server cryptographically generates a ticket in response to an end user's request for access to a file. The ticket is based, at least in part, on a time at or near when the ticket is generated. In certain embodiments, the ticket is based on additional information, including, for example, a security time interval, or an identifier of the end user.

[007] Prior to a media server providing access to the requested file, the media server generates an authorization ticket, preferably using the same cryptographic algorithm as the web server. The media server authorization ticket is based, at least in part, on a time at or near when the media server receives the request for access to the file. The media server determines whether to grant access to the file by comparing the ticket, as generated by the web server, to the ticket, as generated by the media server.

[008] In one embodiment, if the tickets do not match, then the time at which the web server generated the ticket differs from the time at which the media server generated the ticket by more than a predetermined amount, and the ticket can be logically thought to have "expired." Accordingly, the media server does not grant access to the media content. If the tickets match, then the tickets were generated within an authorized time interval, and the media server grants the end user access to the requested media content.

BRIEF DESCRIPTION OF THE DRAWINGS

[009] Fig. 1 is a schematic illustrating the system according to one embodiment of the present invention;

[0010] Fig. 2 is a schematic illustrating the database according to one embodiment of the present invention;

[0011] Fig. 3 is a schematic illustrating the workflow according to one embodiment of the present invention;

[0012] Fig. 4 is a flowchart illustrating the process of generating a ticket according to one embodiment of the present invention;

[0013] Fig. 5 is a flowchart illustrating the process of determining whether to provide access to a item of streaming media content according to one embodiment of the present invention;

[0014] Fig. 6 is a schematic illustrating the system according to an alternate embodiment of the present invention;

[0015] Fig. 7 is a schematic illustrating the database according to an alternate embodiment of the present invention; and

[0016] Fig. 8 is a schematic illustrating the workflow according to an alternate embodiment of the present invention.

DETAILED DESCRIPTION OF PREFERRED EMBODIMENTS

[0017] Certain preferred embodiments of the present invention will now be described with reference to the drawings. Although the invention for controlling access to content is described in the context of controlling access to streaming media files, it is to be understood that the present invention is applicable to all types of media or files. Furthermore, one skilled in the art will recognize that although the embodiments discussed herein relate to on-demand streaming media, the present embodiment is also applicable to live streaming media.

[0018] In general, the system of the present embodiment includes end user processors 102, a streaming media server 104 and a web server 106 having a content management (CM) database 108, all of which are coupled to the Internet. The end user processors 102 include an Internet browser, such as that provided by the Microsoft Corporation under the name INTERNET EXPLORER or provided by Netscape Communications under the name NETSCAPE NAVIGATOR, and a streaming media player, such as that provided by the Microsoft Corporation under the name WINDOWS MEDIA PLAYER or that provided by Real Networks, Inc. under the name REALPLAYER. The web server 106, provides a website accessible by the end users 102. The website, in turn, includes links that can be activated by the end users 102 for accessing streaming media content residing on the streaming media server 104.

[0019] It is to be understood that the present invention may be implemented utilizing any number of computer technologies. For example, although the present embodiments relate to providing access to content via the Internet, the present invention may be utilized over any computer network, including, for example, a wide area network. Similarly, the end user processors 102 may be any device that may be coupled to the network, including, for example, personal digital assistants, web-enabled cellular telephones, hard-wired telephones that dial into the network, mobile computers, personal computers, Internet appliances and the

like. Furthermore, the servers described herein may be of any type, running any software, and the software modules, objects and plug-ins described herein may be written in any programming language. Lastly, the database and storage devices described herein may utilize any storage technology, including, for example, local computer memory, network attached storage, and any known storage medium, such as magnetic or optical.

[0020] An exemplary representation of the CM database 108 is illustrated in Fig. 2. As shown, the database 108 includes information universally applicable to all items of streaming content and several tables of related data. The universal information 202 includes a security key, a security time interval and the name ("Hostname") of the streaming media server 104 on which the content resides. The security key and the security interval are used in authorizing end users 102 to access the content and, therefore, are preferably maintained in secret and set by the owner of the content. The security key and security interval are used for controlling access to all content, although in alternate embodiments each content file has its own security key and security interval associated therewith.

[0021] The CM database 108 further includes a series of tables containing content or stream identifying information. More specifically, the Streams Table 204 includes a record for each item of streaming content, as identified by a unique stream identifier (ID). Furthermore, each record includes: the stream details that describe the content file, including, for example, the creation date of the content file, a description of the file, an identification of whether the content is audio and or video, the platform to which the content relates, the date on which the content was last modified, any codec necessary for viewing of the content, the length and size of the content, the expiration date (if any) of the content, the stream type, such as .asf or .rm, title of the content, author of the content, status of the content, copyright notice for the content, bit rate of the content and the like. Each record also includes: the prefix used to generate a link to the media server 104 ("URL Prefix"); and the name of the content file

("Filename"), as stored on the streaming media server 104. It should be understood that the filename may point to an actual path on storage coupled to a streaming media server 104 for on-demand content or the filename may point to an alias, port or channel for a live stream.

[0022] The database 108 also includes tables containing "playlist" information. A client's playlist is generally a group of one or more content files logically associated for the purpose of being made available as a group. Each content file identified as part of a playlist can also be made available individually. Such playlist information is contained within the Playlist Table 208 and the Playlist Streams Table 210. In general, the Playlist Table 208 includes records identifying each playlist, as identified by a playlist ID. Each record further includes playlist details, including, for example, the playlist format (such as Windows Media Player or RealPlayer), the playlist description, the playlist name, and the like, and the authorized user group ID for the playlist.

[0023] The authorized user group ID corresponds to a group of end users 102 that are authorized to view the particular playlist. More specifically, the database 108 further includes an Authorized User Table 206 that correlates each end user 102, as identified by a unique end user ID, to one or more authorized user group IDs. In order for an end user 102 to view a playlist, the end user 102 must be identified as part of the authorized user group ID for that content file. In certain alternate embodiments, no authorized group ID is used, while in other alternate embodiments each content file has an authorized group ID associated therewith.

[0024] The Playlist Streams Table 210 includes records correlating each playlist, as identified by the playlist ID, with the constituent content files, as identified by stream ID. Each record also contains the information indicative of the order of each content file in the playlist ("Sort Order").

[0025] Having described the components utilized in the present embodiment, the process of controlling access to the streaming media content will now be described. By way of overview, an authorization software component located on the web server 106 generates a hash value or "ticket" based upon public key information, private key information and the then current time. The public key is a unique identifier for the streaming content requested by the end-user 102 and the end user's user ID. The private key includes a security key and security time interval set by the owner of content.

[0026] The streaming media server 104 on which the requested content resides receives the stream request, which includes the public key, and the ticket as generated by the web server 106. The streaming media server 104 proceeds to use locally stored private key information to generate its own version of the ticket. The streaming media server 104 either denies or provides access to the requested streaming media content based on a comparison of the tickets as generated by the streaming media server 104 and web server 106.

[0027] The process of controlling access will now be described in greater detail with reference to the workflow diagram of Fig. 3 and the flow charts of Figs. 4 and 5. In the present example, the end user 102 requests access to an individual streaming media content file. Initially, the web server 106 provides a web page requesting the end user to log in to an authorization application and presenting the end user with the option of viewing certain streaming media. Step 302. For example, such a page may include a form requesting the end user to select for viewing one of several content files, each of which has its own stream request link, to provide its end user ID (which the owner of the content previously assigned and provided to the end user) and to provide a credit card number so that the end user may be charged for access to the selected content. In an alternate embodiment, the end user previously registered with the owner of the content by providing the end user's contact

information and billing information, which the owner stores in a table in the database along with the assigned end user ID.

[0028] In response to the web page, the end user provides the end-user's user ID and activates a link, thereby logging into the authorization application and requesting access to the particular streaming media content file associated with the link. Step 304. An exemplary stream request, wherein the stream ID is represented by "123456," is as follows:

<A href http://webserver.company.com/getstream.asp?ID=123456>

[0029] In the present embodiment the authentication application is a ".dll" software component residing on the web server 106. However, one skilled in the art will recognize that any other programming language or technology, such as, for example, an active server page (ASP) or servlet, could be used to implement the functionality described herein. Irrespective of the particular programming technology, it is preferable that the authentication application run on the web server 106 to alleviate any processing bottlenecks on the end user processor 102.

[0030] Once the end user logs into the authentication application and the web server 106 receives the stream request and the end user ID from the end user, the web server 106 continues by dynamically generating the authentication ticket and dynamically generate a link to the selected content file. More specifically, under control of the authentication application, the web server 106 issues a request to the database 108 for the private key for use in generating the authorization ticket. Step 306. The web server 106 issues a database query to retrieve from the CM database 108 the private key, comprising the security key and security interval associated with the requested content file. In response, the CM database 108 returns the private key to the web server 106. Step 308.

[0031] Having obtained the private key from the database 108, the web server 106 generates the ticket. Step 310. As described more fully with reference to Fig. 4, the web server 106

utilizes the private key, stream ID, end user ID, the current time and a hash algorithm to generate the ticket. In the present embodiment, the web server 106 can use the stream ID to generate the ticket because the stream ID of the requested content is included in the stream request link activated by the end user in step 304. In alternate embodiments, however, the stream request provided by the end user includes unique identifying information other than the stream ID, such as, for example, the title, author and/or filename of the content. In such an embodiment, the web server 106 searches the Streams Table 204 and retrieves the stream ID based on the identifying information contained in the stream request. In yet another alternate embodiment, the stream request includes a unique identifier other than the stream ID, such as the filename or path, which the system uses to generate the ticket.

[0032] Once the ticket is generated, the web server 106 generates the link to the requested content on the media server 104. More specifically, based on the illustrative stream request shown above, the media player residing at the end user processor 102 makes a call to "webserver.company.com" (i.e., the web server 106) that will execute the "getstream.asp" program for dynamically generating the link to the media server 104. Step 312. One skilled in the art will recognize that although the "getstream" application has an Active Server Page (or ASP) extension, it is not necessary to use ASP technologies. Rather, any programming or scripting language or technology, such as a ".dll" component, could be used to provide the desired functionality. As with the authentication application, it is preferred, however, that the program run on the server side so as to alleviate any processing bottlenecks at the end user processor 102. The "getstream.asp" program functions to cause the web server 106 to make a call to the CM database 108 to retrieve the data necessary to dynamically generate the link to the media server 104. More specifically, the web server 106 retrieves the Hostname from the Universal Info Table 202 and the URL Prefix and Filename from the Streams Table 204. The "getstream.asp" program also appends the stream ID, the ticket and the end user ID to the end

of the link. The web server 106 then returns the link to the media player at the end user processor 102. Step 314.

[0033] An illustrative link to the media file is as follows, wherein: the URL Prefix is requested by "mms://"; the Hostname is represented by "mediaserver.company.com"; the Filename is represented by "stream1.asf"; the stream ID of the requested item of content is represented by "123456"; the ticket is represented by "uvw123xyz"; and the end user ID is represented by "abc123def".

<REF href="mms://mediaserver.company.com/stream1.asf?ID=123456&TICKET=
uvw123xyz& USER_ID=abc123def">

[0034] Having received the link, the end-user processor 102 proceeds to request the streaming media content. Step 316. More specifically, the media player residing on the end user processor 102 makes a call to "mediaserver.company.com" (i.e., the streaming media server 104), as identified in the link. As part of the call, the media player provides the streaming media server 104 with the copy of the requested content's stream ID, the ticket generated by the web server 106 and the end user ID.

[0035] Having received the link, which includes the stream ID, the end user ID and the ticket, the streaming media server 104 proceeds to determine whether or not to grant the end user access to the requested content. Step 318. As described in greater detail below with reference to Fig. 5, the streaming media server 104 determines whether or not to grant access by independently generating a ticket based on locally stored private key information and the stream ID and end user ID contained in the link. In general, if the ticket generated by the streaming media server 104 matches the ticket as generated by the web server 106, the streaming media server 104 provides the requested streaming media content to the end user processor 102. Step 320.

[0036] The process of generating the ticket by the web server 106 will now be described in greater detail with reference to Fig. 4. As noted above, the ticket generation process is

preferably performed by an authorization software plug-in residing at the web server 106. In the present embodiment, the process beings with the web server 106 receiving the stream request, including the stream ID, and the end user ID. Step 402. The web server 106 then proceeds to access the database 108 to retrieve the private key information associated with the requested stream ID. Step 406. Such private key information includes the universal security key and the security interval. In an alternate embodiment, each stream has its own security key and security interval stored as fields in the Streams Table 204, which the web server 106 retrieves based on the stream ID contained in the stream request.

[0037] As noted above, the web server 106 also uses the current time to generate the ticket. More specifically, the web server 106 calculates the current time and rounds the time down to the nearest multiple of the security interval. Step 410. The present embodiment utilizes Universal Coordinated Time (UTC) in seconds, as generated by the C programming language standard library function "time ( )". Exemplary Perl programming code for generating the time as rounded down to the nearest multiple of the security interval (represented by the variable "$time") is as follows, wherein the variable "$interval" corresponds to the security interval, which equals 15 minutes.

```
#
# example of 15 minute ticket expiration/security interval
#
$interval = 15 * 60
$time = int (time( ) / $interval) * $interval;
```

[0038] By way of example, if the current time was May 31, 2000 at 2:16:07 pm, Central Time, the function "time ( )" returns a value of approximately "959800567". Rounding down this UTC value to the nearest 15 minute interval results in a value of approximately "959800500," which represents a time of May 31, 200 at 2:15:00 pm Central Time.

[0039] It is to be understood that the foregoing exemplary code may be modified and still be within the scope of the present invention. For example, the security interval need not be in minutes; the interval may be represented in other units of time so long as an appropriate conversion is performed so that the interval is represented in the unit of time utilized by the "time ( )" function. Furthermore, in alternate embodiments the current time is based on a standard other than UTC. In one such embodiment, the time standard is unique to the web server 106 and streaming media server 104. It is also to be understood that it is within the scope of the present invention to have the end user processor 102 calculate the time and pass the value to the web server 106 for use in generating the authorization ticket. In still further alternate embodiments, the security interval is selected so that the standard time is simply truncated to a desired number of digits.

[0040] Once the web server 106 has the input values to the hash algorithm—the public key information, private key information, and the time value—the web server 106 generates the input string to the hash algorithm. Step 414. In the present embodiment, the hash algorithm is the "MD 5" message digest algorithm. Also in the present embodiment, the media server 104 and web server 106 utilize the same algorithm.

[0041] It is to be understood that it is within the scope of the present invention to utilize essentially any hash or cryptographic algorithm to generate the ticket. Furthermore, the two servers generating the tickets (in the foregoing embodiment, the web server 106 and the streaming media server 104) preferably generate the same ticket based on the same inputs or tickets within a known deviation of each other based on the same inputs. In alternate embodiments one of a plurality of potential algorithms are used to increase security. By way of example, such embodiments use one randomly selected algorithm from a plurality of potential algorithms or can select one of a plurality of algorithms based on the requested content, the date or time of the request, the particular end user, the entity owning the content,

and the like. In such embodiments, the system passes to the media server an indication of the algorithm used by the web server, or the media server includes logic that causes it to select and use the same algorithm utilized by the web server.

[0042] Any arrangement of the input values may be used as the input string so long as the input string is valid for the particular hash algorithm being used and so long as the streaming media server 104 knows the arrangement of the input string. In the present embodiment the following predetermined arrangement is used, wherein "T" represents a digit of the time value, "K" represents an alphanumerical character in the security key, "S" represents a digit of the stream ID (including any necessary leading padding characters) and "U" represents an alphanumerical character of the end user ID (including any necessary leading padding characters).

TTTTTTTTTTKKKKKKKKKKSSSSSSSSSSUUUUUUUUUU

In alternate embodiments input strings may be of different lengths.

[0043] Having generated the hash algorithm input string, the web server 106 applies the hash algorithm to the input string, thereby generating the ticket. Step 418.

[0044] The process of the streaming media server 104 determining whether to grant access to the requested content stream will now be discussed with reference to Fig. 5. As an initial matter, it should be noted that, although not required, the media server of the present embodiment 104 generates three authentication tickets, each based on a different time value, for use in determining whether to grant access. Furthermore, as with the web server functionality, it is preferable that the process of determining whether to grant access is implemented in an authorization software component residing on the media server 104.

[0045] In determining whether to grant access, the streaming media server 104 first receives the stream request, including the stream ID, end user ID and ticket, from the media player residing on the end user's processor 102. Step 502. Once the stream request is received, the

media server 104 generates the input string to the hash algorithm. In this regard, the media server 104 retrieves from local memory the private key information, namely the security key and security interval. Step 506. Preferably, the media server 104 stores the private key information in local memory, however, in alternate embodiments the media server 104 stores the information in an active directory tree accessed by, for example, Light-Weight Directory Access Protocol provided by the Microsoft Corporation, or in a remote database. In still another alternate embodiment, the media server 104 retrieves the private key information by accessing the database 108 via a network connection, such as Local Area Network (LAN).

[0046] As did the web server 106, the media server 104 also calculates the current time and rounds the time down (i.e., earlier in time) to the nearest multiple of the security interval. Step 510. Unlike the web server 106, however, the streaming media server 104 also calculates a second time value equal to the current time rounded down to the next nearest multiple of the security interval below (i.e., earlier than) the first time value calculated by the media server 104. Step 510. The media server 104 further calculates a third time value equal to the current time rounded up (i.e., later in time) to the nearest multiple of the security interval. Step 510.

[0047] The media server 104 then uses the retrieved private key information, the received public key information and the three time values to generate three corresponding hash input strings. Step 514. The media server 104 then applies each of the three input strings to the hash algorithm, thereby generating three tickets. Step 518.

[0048] Having independently generated the tickets, the media server 104 then determines whether any of the tickets generated by the media server 104 match the ticket as generated by the web server 106. Step 522. If the tickets do not match, then it is likely that the stream request is not authentic and/or has expired (i.e., was generated by the media server 104 at a

time outside of the security interval as measured from the time of the user's request). Accordingly, the media server 104 denies access to the requested content. Step 526

[0049] If the tickets do match, then it is likely that the stream request is both authentic and within the security interval. However, prior to granting access, the media server 104 first determines whether the end user has already requested access to and viewed the same content. Step 530. The media server 104 maintains, preferably in local memory, a list of end user IDs and corresponding stream IDs to which the end user has been granted access. To determine whether the end user has already viewed the requested content, the media server 104 accesses memory to determine whether the received end user ID and stream ID have previously been stored. If the end user ID and stream ID have previously been stored, then the end user is denied access to the requested content. Step 530.

[0050] If the received end user ID and stream ID have not previously been stored, the media server 104 proceeds to store the end user ID and stream ID in memory (step 534) and provides the end user access to the content. Step 538. As such, storing the end user ID and stream ID provides an added, optional level of security protection that prevents end users from sharing the link pointing to the requested content with others.

[0051] It is to be understood that the use of three tickets is preferable to account for a lack of synchronization between the local time of the web server 106 and the local time of the media server 104. Furthermore, in certain circumstances the first ticket generated by the media server 104 (i.e., based on the current time rounded down to the nearest multiple of the security interval) will not match the first ticket generated by the web server 106 even though the end user is authorized. For example, given a security interval of 15 minutes, if the web server 106 generates the ticket at 12:14:00 pm and the media server 104 generates its first ticket at 12:16:00 pm, on the same day in the same time zone, the tickets will not match even though the request is within the security interval. The web server will generate the ticket

based on a time value corresponding to 12:00:00 pm, while the media server 104 will generate a ticket based on a time value corresponding to 12:15:00 pm. Accordingly, in the present embodiment the media server 104 generates the second ticket based on the then current time rounded down to the next nearest multiple of the security interval; in the present example, a time value corresponding to 12:00:00 pm. As such, the second ticket would match the ticket as generated by the web server 106. Similarly, it is possible for access to be granted to an end user after the security interval has elapsed. Thus, in the present embodiment the security interval should be selected to account for the use of multiple tickets. Preferably, the web server 106 and the media server 104 have clocks synchronized to within about one-half of the security interval.

[0052] It is to be understood that it is also within the scope of the present invention for the media server 104 to generate one or more different tickets as an alternative to the three tickets in the foregoing embodiment. Furthermore, although the foregoing embodiment describes the tickets as being generated together, in parallel, it is within the scope of the invention for the media server 104 to generate and/or compare the tickets, one after another, in serial. It is also to be understood that the time values may be generated in any of a number of ways, including, for example, by simply adding or subtracting the security interval from the first time value calculated by the media server 104.

[0053] In an alternate embodiment, another level of security may be provided. Specifically, if the ticket generated by the web server 106 matches one of the tickets generated by the media server 104, then the media server 104 proceeds to determine whether the same ticket has been previously generated. The media server 104 maintains a list of tickets for which access has been granted. Such a list logically represents all "used" tickets. If the matched ticket is not on the list of "used" tickets, then the media server 104 grants access, providing the requested content to the media player residing at the end user's processor 102. As part of

granting access, the media server 104 also updates the listing of "used" tickets. If the matched ticket is on the list of used tickets, then the media server 104 denies access and provides an appropriate message to the requesting end user. By tracking the used tickets, the system prevents an authorized end user from sharing the streaming request received from the web server 106 with others.

[0054] It is also to be understood that it is within the scope of the present invention to use error calculations in determining whether to grant access. For example, one error calculation involves the media server 104 generating one or more additional tickets based on the current time plus and/or minus an error interval, such as, for example, a predetermined time period (e.g., 15 minutes, 30 minutes, etc.), a set percentage of the applicable security interval (e.g., 50%, 125%, etc.) or some other error calculation. Such error calculations may be used as an alternative to the second or third time values in the foregoing embodiment or in addition thereto.

[0055] In alternate embodiments the web server 106 and media server 104 generate tickets by calculating the time value differently than the foregoing embodiment. In one exemplary embodiment, the web server 106 and media server 104 calculate the current time and round it down to or up to a multiple of some interval other than the security interval. In one such an embodiment where the security interval is 15 minutes, the web server 106 generates the ticket based on the current time rounded down to the nearest interval of 5 minutes. The streaming media server 104, in turn, generates a ticket based on the current time rounded down to the same interval of 5 minutes. If the tickets do not match, the media server 104 proceeds to generate a ticket based on the time rounded down to the next lower interval. The media server continues to generate tickets based on the next lower interval for a set number of times or until the web server and media server tickets match. Preferably, the media server 104 repeatedly generates new tickets based on time intervals the sum of which span at least the

security interval. In the present example, the media server 104 generates at least three tickets, each an interval of 5 minutes, for a total of 15 minutes.

[0056] It is to be understood that it is within the scope of the present invention to entirely omit use of the end user ID in the authorization process or to use the end user ID in a manner different from that described above. For example, in an alternate embodiment the end user ID is not used as part of the input string to the hash algorithm. Instead, the database 108 also includes tables for tracking which end users have requested access to the content. Such an embodiment includes a Viewing User (Streams) Table that contains records correlating content, as identified by stream ID, with end users, as identified by end user IDs, that have accessed or viewed the content stream. The embodiment similarly includes a Viewing User (Playlists) Table that contains records correlating playlists, as identified by playlist ID, with end users, as identified by end user IDs, that have accessed or viewed the playlist. Before generating the authorization ticket, the web server checks the appropriate Viewing User Table to determine whether the same end user has requested access to a particular stream or playlist. In the event an end user has previously requested access, the web server either denies access or provides a web page to the end user indicating that the end user will be charged again for the subsequent access. The tables are automatically cleared after a period of time, such as the security interval or some period in excess thereof.

[0057] It is to be understood that the present invention may also be embodied in relatively more complex systems, for example, ones in which a service provider operates web servers, streaming media servers, and playlist servers, on behalf of its clients—the owners of the content. One such an embodiment will now be described with reference to Figs. 6-8. It will be understood by one skilled in the art that much of the functionality of the present embodiment is the same as that of the embodiment of Figure 3 and, as such, can be implemented by any of the same technologies.

[0058] As shown in Fig. 6, the system includes several components similar to those of the embodiment of Fig. 1, including end user processors 602, one or more streaming media servers 604, and one or more web servers 606, including a database 608, all of which are couple to the Internet or other network. Additionally, the system of the present embodiment also includes a playlist server 610 that is also operated by the service provider. Preferably, the web server 606, streaming media servers 604, including the database 608, and playlist server 610 are connected to the service provider's network, such as a local area network (LAN) or wide area network (WAN), and the Internet.

[0059] In general, the database 608 includes the same information contained in the database of the embodiment of Fig. 2, however, the information is stored on a client account-by-client account basis. As shown in Fig. 7, the database 608 includes an Account Table 702 that includes a record for each client, as identified by an Account ID. Each record further includes: client-identifying information ("Client Info"), such as client name, address, billing information, and the like; an indication as to whether or not the client's content is secure ("Allows Secure"); the client's security key ("Security Key"); and security interval ("Security Interval").

[0060] As with the embodiment of Fig. 2, the present database 608 also includes a Streams Table 704, which includes stream identifying information for each content file, as identified by stream ID, an Authorized User Table 706,which associates end user IDs with Authorized User Group IDs, a Playlist Table 708, which contains playlist identifying information for each playlist, as identified by playlist ID, and a Playlist Streams Table 710, which identifies the stream IDs associated with a given playlist ID. In addition to the information fields described in connection with the database of Fig. 2, the present Streams Table 704 and Playlist Table 708 further include a field identifying the Account ID associated with each content file and each playlist, respectively.

[0061] The present database 608 also includes a Streams-Server Table 712 that contains a record for each content file, as specified by stream ID, identifying the Hostname of the particular streaming media server 104 on which the content file resides. As with the embodiment of Fig. 2, the Hostname is the DNS name of the media server 104.

[0062] The operation of the present embodiment will now be described with reference to the work flow diagram of Fig. 8. For purposes of the present example, the end user requests access to a playlist having one item of secure content. Initially, the web server 606 provides a web page requesting the end user to log in to an authorization application and presenting the end user with the option of viewing certain streaming media. Step 802. As with the embodiment of Fig. 3, an exemplary web page may include a form requesting the end user to select a particular content file by activating a link, provide an end user ID and provide billing information. In response to the web page, the end user provides the end-user's user ID and credit card information and activates the stream request link, thereby requesting access to a particular streaming media content file. An exemplary stream request link, where the playlist ID is "789000", is as follows:

<A href "http://playlistserver.company.com/makeplaylist.dll?ID=789000">

[0063] When the end user activates the stream request link, a programming script running on the end user processor 602 causes the stream request link and the end user ID to be sent to the web server 606. Step 804. One skilled in the art will recognize that the end user script may be implemented in essentially any programming language, including, for example, C++, Perl, Visual Basic, Java and the like. In the present embodiment, the script is a Java script and is running in conjunction with the end user's web browser.

[0064] Once the web server 606 receives the stream request from the script, the web server 606, under the direction of an authorization software plug-in, generates the ticket. In this regard, the web server 606 issues a request to the database 608 for the private key (in the

present embodiment the security key and security interval associated with the requested playlist) for use in generating the authorization ticket. Step 806. In response, the database 608 returns the private key to the web server 606. Step 808.

[0065] Having obtained the private key from the database 608, the web server 606 generates the ticket as described above with reference to Fig. 4. Using the playlist ID instead of the streamID. Step 810. As described therein, the web server 606 applies the private key, stream ID, end user ID and the time values to a hash algorithm to generate the ticket. The web server then returns the ticket and the end user ID to the web browser running on the end user processor 602. Step 812.

[0066] Having received the ticket, the script running on the end user processor 102 appends the information to the end of stream request link. Step 814. An exemplary link, wherein the playlist ID is represented by "789000," the ticket is represented by "uvw123xyz," and the end user ID is represented by "abc123def," is as follows:

<A href "http://playlistserver.company.com/makeplaylist?ID=789000&TICKET=uvw123xyz return&USER_ID=abc123def">

[0067] The script running on the end user processor 602 then causes a call to be made to the playlist server 610, as identified in the stream request link by the Hostname "playlistserver.company.com." Step 816. Accordingly, the playlist server 810 is provided with the link, playlist ID, ticket and user ID. Under control of the "makeplaylist.dll" object, the playlist server 610 generates a redirector file, such as an ASX file where the content is in the Windows Media format. Step 818. The "makeplaylist" program may be implemented using any of a number of programs or technologies, including, for example, an ASP. The redirector file contains a link to the requested content, along with the ticket and public key (i.e., stream ID and end user ID). To generate the redirector file, the playlist server 610 accesses the database 608 to retrieve the stream ID of content file comprising the playlist and

the information necessary to link to the content file, including the Hostname, URL Prefix and Filename, associated with the stream ID.

[0068] In an alternate embodiment, no end user script is utilized to append the ticket to the stream request. Instead, when the end user provides its end user ID and activates the stream request link (in step 804), the authentication application running on the web server 606 generates the ticket, appends the ticket and end user ID to the stream request link, and directly makes the call to the playlist server 610 to create the redirector file. Because the web server 606 also passes to the playlist server 610 information identifying the media player on the end user processor 602, the playlist server 610 forwards the redirector file to the media player (thereby obviating steps 812, 814 and 816).

[0069] The playlist server 610 then passes the ASX redirector file to the media player at the end user process of 602. Step 820.For purposes of the present example, the ASX file is as follows, wherein the URL Prefix is represented by "mms://"; the Hostname of the appropriate media server 604 is represented by "mediaserver.company.com"; the Filename is represented by "stream1.asf"; the requested item of content stream ID is represented by "123456"; the ticket is represented by "uvw123xyz" and the end user ID is represented by "abc123def"; and:

```
<ASX>
   <ENTRY>
      <REF href="mms://mediaserver.company.com/stream1.asf?ID=123456&TICKET=
         uvw123xyz& USER_ID=abc123def">
   </ENTRY>
</ASX>
```

[0070] The redirector file may include other information, such as metadata for the content file, or other, non-secure files, such as advertisements.

[0071] Having received the ASX file, the end-user processor 602 proceeds to request the streaming media content. More specifically, the media player makes a call to

"mediaserver.company.com" (i.e., the streaming media server 604), as identified in the ASX file. Step 822. Once the call is made, the media player provides the streaming media server 604 with the copy of the requested content's stream ID, the ticket generated by the web server 606 and the end user ID.

[0072] In response to the media player's call, the streaming media server 604 proceeds to determine whether or not to grant the end-user access to the requested content. Step 824. The streaming media server 604 determines whether or not to grant access by independently generating one or more authentication tickets and comparing the tickets to the ticket generated by the web server 606. The process of generating and comparing the authorization tickets is achieved in the same manner as described with reference to Fig. 5, using the playlist ID instead of a stream ID. If a ticket generated by the media server 604 matches the ticket generated by the web server 606, the media server 604 grants the end user access to the requested content. Step 824.

[0073] It is to be understood that although the foregoing embodiments utilize a private key comprising both a security key and a security interval, it is within the scope of the present invention to utilize more or less information as the private key. For example, in alternate embodiments, no security key is used and in other embodiments, additional information is included in the private key, including, for example, a client's user name and password. Similarly, it is within the scope of the present invention to utilize a public key comprising information other than the stream ID and end user ID. For example, other content file identifying information may be used, including, for example, the file path name. Additionally, the end user ID may be omitted from the public key information in certain embodiments. In still other embodiments, the public key information includes additional information, such as the title or other stream detail of the request content file.

[0074] It is also to be understood that the functionality described as being provided by the web servers and the streaming media servers may be implemented on other devices associated therewith. For example, in certain embodiments of the present invention, the streaming media server has an associated application server coupled thereto, which implements all or part of the process of denying or granting access to content. Similarly, it is within the scope of the present invention to associate an application server with the web server to provide some or all of the functionality of the web server, including, for example, the process of generating the authorization ticket. As such, reference to a particular server is meant to include other associated servers or processors coupled to the referenced server.

[0075] It is also to be understood that the authorization tickets need not be generated at precise times. For example, the ticket is generated by the web server may be based on the time when the end user activates the stream request link, when the web server receives the private key information from the database, or any other time near the activation of the stream request. Similarly, the media server may generate authorization, for example, when the call is made from the media player, after the private key information is retrieved, or any other time near the time a call is made for the content. Furthermore, where the media server generates multiple tickets, the tickets may be based on different times or the same time. Accordingly, reference to time or the current time is meant to refer to a range and not a precise time.

[0076] Although the foregoing exemplary embodiments have been discussed in the context of controlling access to a single item of content, those skilled in the art will understand that any of the foregoing embodiments may be utilized to control access to a playlist comprising multiple secure content files. One exemplary embodiment for controlling access to a playlist will now be described with reference to the embodiment of Figs. 6-8. Such an embodiment operates in accordance with foregoing description, with the modifications noted below. In

general, the web server 606 generates a ticket for each content stream contained in the playlist based on each stream's stream ID.

[0077] The media player on the end user processor 602 passes the stream request, which includes the playlist ID, to the playlist processor 610. The playlist processor 610, in turn, generates the redirector file and returns the redirector file to the media player. The "makeplaylist.dll" object uses the playlist ID, "789000" in the present example, to construct the appropriate redirector file. More specifically, the playlist server 610 accesses the Playlist Table 708 and the Playlist Streams Table 710 to determine which content files are part of the requested playlist and the order of the content files in the playlist. The content files' filenames are retrieved from the Streams Table 704. A script running on the end user processor 602 then appends the stream IDs, tickets and end user ID to the URL linking to the corresponding content stream. In the present embodiment, all content streams are located on the same media server 604, as identified in the Streams-Server Table 712.

[0078] An exemplary ASX redirector file, including the stream IDs, tickets and end user ID appended to the URL link for the corresponding content stream, is as follows:

```
<ASX>
    <ENTRY>
        <REF href="mms://mediaserver.company.com/stream1.asf?ID=123456&TICKET=
            abc111xyz&USER_ID=abc123def">
        <REF href="mms://mediaserver.company.com/stream2.asf?ID=234567&TICKET=
            def222xyz&USER_ID=abc123def">
        <REF href="mms://mediaserver.company.com/stream3.asf?ID=345678&TICKET=
            ghi333xyz&USER_ID=abc123def">
    </ENTRY>
</ASX>
```

[0079] The media player then makes a series of calls to the streaming media server 604, one for each of the URL links contained in the redirector file. More specifically, the media player first makes a call to the media server 604 for access to the first content stream (in the present

example, having stream ID 123456). In response to the call and as generally described above with reference to Fig. 5, the media server 604 independently generates a ticket and determines whether to grant access to the content. If access is not granted, the end user is notified as such. On the other hand, if the media server grants the end user access to the first content stream, then the media player proceeds to makes calls to the media server 604 for the remaining content streams in the playlist. With each call, the media server 604 proceeds with authorizing or denying access to the requested content stream.

[0080] It should be understood that in such an embodiment, however, it is preferable for each content stream to have an individual security interval that accounts for the total duration of the content streams played prior to the stream in the playlist. For example, in a playlist containing three content streams, each of which is five minutes in duration (as identified in Stream Details fields of the Streams Table 704), the security interval for the second stream may be five minutes longer than that for the first stream, and the security interval for the third content stream may be ten minutes longer than the interval for the first stream. By accounting for the duration of each stream in the playlist, the system helps prevent an authorized end user from receiving access to the first content stream in the playlist but not to a subsequent content stream because the ticket has expired. The security intervals may also account for any non-secure content, such as advertisements, contained in the playlist.

[0081] Other alternate embodiments control access to a playlist containing multiple secure content streams by generating a ticket based on the playlist ID. One such an embodiment operates in accordance with the description of the system of Figs. 6-8, with the modifications noted below. In general, once the end user logs in to the authorization application and requests access to a playlist, the web server 606 generates a ticket based on the playlist ID and returns the ticket to the end user processor 602. In response, a script running on the end user processor 602 appends the ticket and end user ID to the stream request link. The

following is an illustrative stream request link having the public key information appended thereto, wherein the playlist ID represented by "789000"; the ticket is represented by "xyz321abc" and the end user ID is represented by "abc123def".

<A href "http://playlistserver.company.com/makeplaylist.dll?PLAYLIST_ID=789000& TICKET=xyz321abc&USER_ID=abc123def">

[0082] The end user processor 602 makes a call to the playlist server 610, as identified by the name "playlistserver.company.com". The playlist server 610, in turn, initiates the "makeplaylist.dll" object residing at the playlist server 610 in order to generate the redirector file. In the present embodiment, all content streams reside on the same media server 604. Unlike prior embodiments, the "makeplaylist.dll" object also appends to the end of the first URL link in the redirector file the filenames for the subsequent secure content streams in the playlist, and only the playlist ID and ticket is appended to each of the subsequent URL links. An exemplary ASX redirector file is as follows, wherein: the playlist includes three Windows Media format content files having the filenames represented by "stream1.asf", "stream2.asf" and "stream3.asf"; the playlist ID is represented by "789000"; the end user ID is represented by "abc123def"; and the ticket is represented by "xyz321abc":

```
<ASX>
    <ENTRY>
        <REF href="mms://mediaserver.company.com/stream1.asf?PLAYLIST_ID=789000&
            TICKET=xyz321abc&USER_ID=abc123def&STREAM=stream2.asf&
            STREAM=stream3.asf">
        <REF href="mms://mediaserver.company.com/stream2.asf?PLAYLIST_ID=789000
            &TICKET=xyz321abc">
        <REF href="mms://mediaserver.company.com/stream3.asf?PLAYLIST_ID=789000
            &TICKET=xyz321abc">
    </ENTRY>
</ASX>
```

[0083] The media player at the end user processor 602 proceeds to make a call to the mediaserver.company.com (i.e., the Hostname of the streaming media server 604) for access

to the first content file. The media server 604 proceeds to generate a ticket based on the playlist ID and to grant or deny access as discussed above with regard to Fig. 5. If the media server 604 grants access to and provides the media player with the first content file in the playlist, the media server 604 creates a record in a locally stored table for the playlist ID and the corresponding ticket, and stores in the record the filenames of the subsequent content streams in the playlist, as contained in the redirector file.

[0084] When the media player subsequently calls for access to the second content stream, the media player provides the playlist ID and ticket to the media server 604. The media server 604, in turn, searches the table for the record identified by the playlist ID and ticket. If the record exists, the media server 604 provides access to the second stream and flags the stream as having been viewed by the end user with the particular ticket. If an unauthorized end user attempts to access the second stream using the same URL link, the media server 604 will deny access because in the record pertaining to the playlist ID and ticket, the second stream has been flagged as having been viewed. The same process is utilized for providing access to the remaining content streams in the playlist. As will be appreciate by one skilled in the art, this embodiment avoids any potential for incorrectly denying access to a subsequent stream in a playlist due to the time delay between granting access to the first stream and such subsequent stream.

[0085] Those skilled in the art will recognize that the method and system of the present invention has many applications, may be implemented in many manners and, as such, is not to be limited by the foregoing exemplary embodiments and examples. Moreover, the scope of the present invention covers conventionally known and future developed variations and modifications to the system components described herein, as would be understood by those skilled in the art.

What is claimed is:

1. A method for controlling access to one or more media files, the method comprising:

a. generating a first authorization ticket based on a first time;

b. transmitting a stream request and the authorization ticket to a server;

c. generating a second authorization ticket based on a second time; and

d. comparing the first authorization ticket and the second authorization ticket to enable a determination whether or not to grant access to the media files.

2. The method of claim 1 wherein step (a) occurs at approximately the first time and step (c) occurs at approximately the second time, the second time later than the first time, the method further comprising:

granting access to the media files if the first authorization ticket matches the second authorization ticket, wherein the first authorization ticket matches the second authorization ticket based on the first time differing from the second time by less than a predetermined amount.

3. The method of claim 1 wherein step (a) occurs at approximately the first time and step (c) occurs at approximately the second time, the second time later than the first time, the method further comprising:

denying access to the media files if the first authorization ticket does not match the second authorization ticket, wherein the first authorization ticket does not match the second authorization ticket based on the first time differing from the second time by more than a predetermined amount.

4. The method of claim 3 wherein the first authorization ticket is further based on a first time value that approximates the first time rounded down to a multiple of a time interval, and the second authorization ticket is further based on a second time value that approximates the second time rounded down to a multiple of the time interval.

5. The method of claim 1 wherein the first authorization ticket and the second authorization ticket are further based on a security key.

6. The method of claim 1 wherein the first authorization ticket and the second authorization ticket are further based on an identifier for the media files.

7. The method of claim 6 wherein the files comprise a playlist and the identifier is an identifier for the playlist.

8. The method of claim 1 wherein the first authorization ticket is generated in response to a request for access to the media files from an end user, and the first authorization ticket and the second authorization ticket are further based on an identifier for the end user.

9. The method of claim 1 further comprising:

generating a third authorization ticket based on a third time; and

wherein step (d) further comprises comparing the first authorization ticket to the third authorization ticket to determine whether to grant access to the media files.

10. The method of claim 9 wherein:

generating the second authorization ticket occurs at approximately the second time, and the second authorization ticket is based on a second time value that approximates the second time rounded down to another multiple of a time interval; and

generating the third authorization ticket occurs at approximately the third time, and the third authorization ticket is based on a third time value that approximates the third time rounded down to a multiple of the time interval, the third time value below the second time value.

11. The method of claim 10 wherein the second time equals the third time.

12. The method of claim 9 wherein:

generating the second authorization ticket occurs at approximately the second time, and the second authorization is based on a second time value that approximates the second time rounded down to a multiple of a time interval; and

generating the third authorization ticket occurs at approximately the third time, and the third authorization ticket is based on a third time value that approximates the third time rounded up to another multiple of the time interval.

13. The method of claim 12 wherein the second time equals the third time.

14. A method for controlling access to one or more media files, the method comprising:

a. receiving a request for the media files;

b. receiving a first authorization ticket associated with the request, the first authorization ticket based on a first time;

c. generating a second authorization ticket based on a second time; and

d. determining whether to grant access to the media file by comparing the first authorization ticket and the second authorization ticket.

15. The method of claim 14 further comprising:

e. denying access to the media files based on the first authorization ticket not matching the second authorization ticket, wherein the first authorization ticket does not match the second authorization ticket based on the first time differing from the second time by a predetermined amount.

16. The method of claim 14 wherein the first authorization ticket is further based on a first time value that approximates the first time rounded down to a multiple of a time interval, and the second authorization ticket is further based on a second time value that approximates the second time rounded down to the multiple of the time interval.

17. The method of claim 14 wherein the first authorization ticket and the second authorization ticket are further based on a security key.

18. The method of claim 14 wherein the first authorization ticket and the second authorization ticket are further based on an identifier for the media files.

19. The method of claim 18 wherein the files comprise a playlist and the identifier is an identifier for the playlist.

20. The method of claim 14 wherein the first authorization ticket is generated in response to a request for access to the media files from an end user, and the first authorization ticket and the second authorization ticket are further based on an identifier for the end user.

21. The method of claim 14 further comprising:

generating a third authorization ticket based on a third time; and

wherein step (d) further comprises comparing the first authorization ticket to the third authorization ticket to determine whether to grant access to the media files.

22. The method claim 14 wherein the request is received from a media player.

23. The method of claim 14 wherein the first authorization ticket is generated by a web server.

24. The method of claim 14 wherein steps (a) through (d) are performed by a media server.

25. A method for an end user to receive access to one or more media files, the method comprising:

a. requesting access to the media files;

b. causing a first authorization ticket to be generated at approximately a first time, the first authorization ticket based on the first time;

c. causing a second authorization ticket to be generated at approximately a second time, the second authorization ticket based on the second time; and

d. receiving access to the media files if the first authorization ticket matches the second authorization ticket, wherein the first authorization ticket matches the second authorization ticket based on the first time differing from the second time by less than a predetermined amount.

26. The method of claim 25 further comprising:

e. supplying an identifier of the end user, wherein the first authorization ticket and second authorization ticket are further based on the identifier.

27. The method of claim 25 wherein step (b) comprises activating a link on a web page.

28. The method of claim 25 wherein step (c) comprises making a call to a media server.

29. The method of claim 25 wherein step (d) comprises a media player receiving the media files by.

30. A system for controlling access one or more to a media files, the system comprising:

a first processor operative with software to generate a first authorization ticket based on a first time;

a second processor operative with software to generate a second ticket, independently of the first processor, based on a second time; and

a third processor operative with software to receive the first authorization ticket and to determine whether to grant access to the media files by comparing the first authorization ticket and the second authorization ticket.

31. The system of claim 30 wherein the second processor is the third processor.

32. The system of claim 30 wherein the first processor is further operative to generate the first authorization ticket based on a first time value and to calculate the first time value by rounding the first time down to a multiple of a time interval, and wherein the second processor is further operative to generate the second authorization ticket based on a second time value and to calculate the second time value by rounding down the second time to the multiple of the time interval.

33. The system of claim 30 wherein the second processor is further operative to generate a third authorization ticket based on a third time, and the third processor is further operative to compare the first authorization ticket and the third authorization ticket.

34. The system of claim 33 wherein the second time equals the third time.

35. The system of claim 33 wherein the second processor is further operative to generate the third authorization ticket based on a third time value and to calculate the third time value by rounding the third time up to a multiple of a time interval.

36. The system of claim 30 wherein the first processor and second processor are further operative to generate the first and second authorization tickets, respectively, based on an identifier of the media files.

37. The system of claim 30 wherein the first processor and second processor are further operative to generate the first and second authorization tickets, respectively, based on a security key.

38. The system of claim 30 wherein the first processor and second processor are further operative to generate the first and second authorization tickets, respectively, based on an identifier of an end user requesting access to the media files.

39. A computer readable medium comprising computer code for instructing one or more processors to:

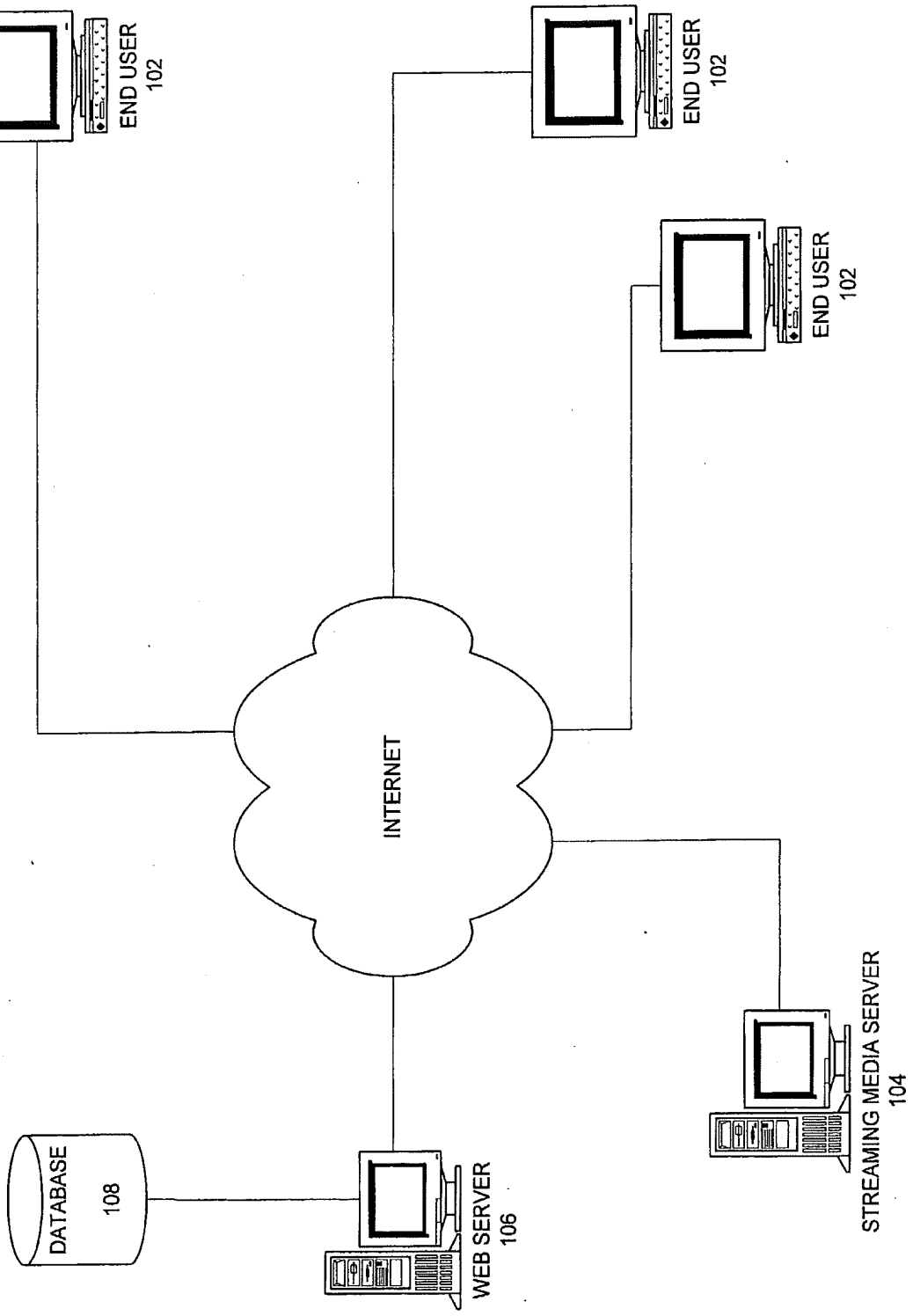
a. receive a first authorization ticket, the first authorization ticket based on a first time, the first authorization ticket associated with a media file;

b. generate a second authorization ticket based on a second time;

c. compare the second authorization ticket to the first authorization ticket; and

d. cause the media file to be transmitted based on comparison of the first authorization ticket to the second authorization ticket.

FIG. 1


END USER
102
END USER
102
END USER
102
INTERNET
DATABASE
108
WEB SERVER
106
STREAMING MEDIA SERVER
104

202
UNIVERSAL INFO TABLE
Security Key
Security Interval
Hostname
204
STREAMS TABLE
Stream ID
Stream Details
URL Prefix
Filename
206
AUTHORIZED USER TABLE
End User ID
Authorized User Group ID
208
PLAYLIST TABLE
Playlist ID
Playlist Details
Authorized User Group ID
210
PLAYLIST STREAMS TABLE
Playlist ID
Stream ID
Sort Order

FIG. 2

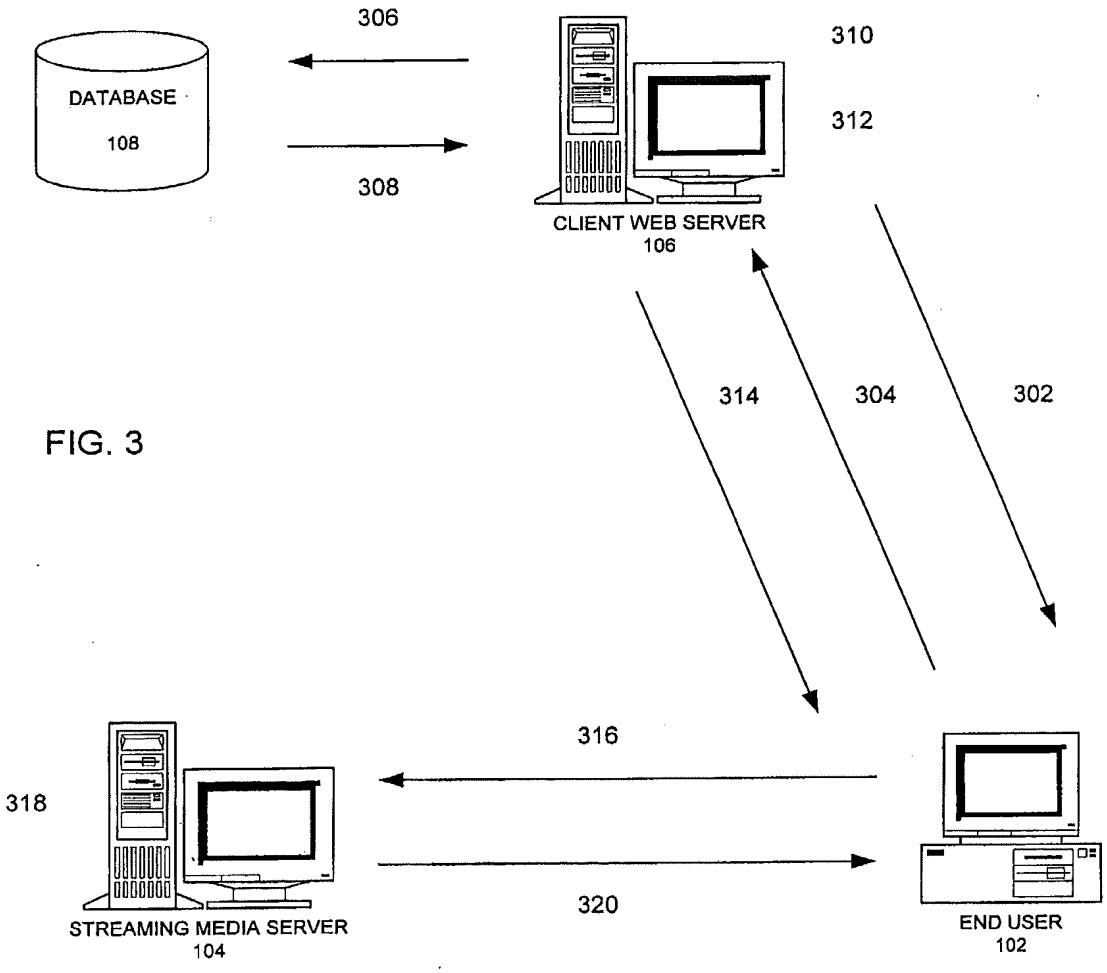
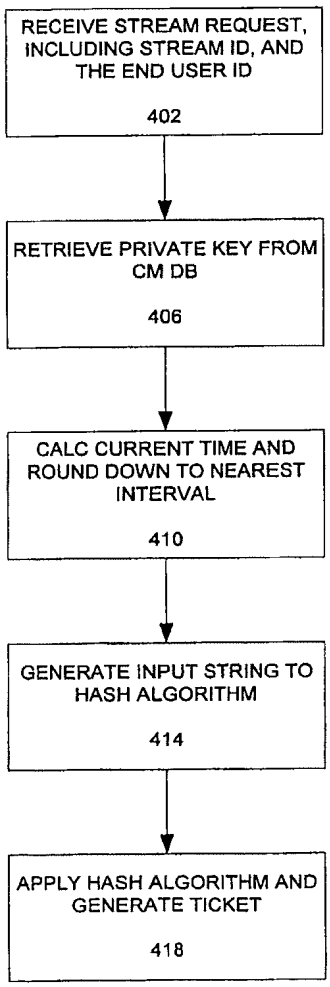

306
DATABASE
108
308
310
312
CLIENT WEB SERVER
106
314
304
302
316
318
320
STREAMING MEDIA SERVER
104
END USER
102

FIG. 3

FIG. 4


RECEIVE STREAM REQUEST, INCLUDING STREAM ID, AND THE END USER ID
402
RETRIEVE PRIVATE KEY FROM CM DB
406
CALC CURRENT TIME AND ROUND DOWN TO NEAREST INTERVAL
410
GENERATE INPUT STRING TO HASH ALGORITHM
414
APPLY HASH ALGORITHM AND GENERATE TICKET
418

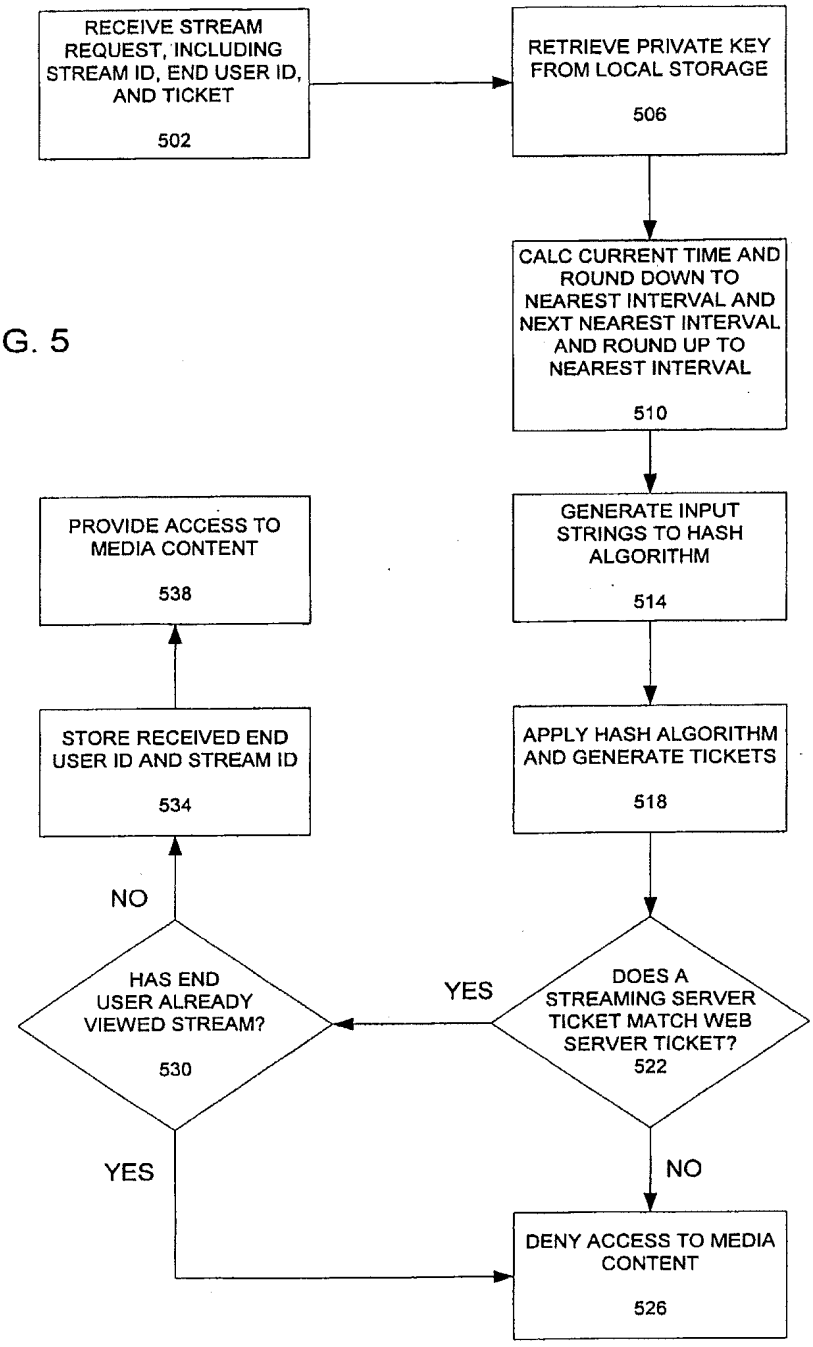

RECEIVE STREAM REQUEST, INCLUDING STREAM ID, END USER ID, AND TICKET
502
RETRIEVE PRIVATE KEY FROM LOCAL STORAGE
506
CALC CURRENT TIME AND ROUND DOWN TO NEAREST INTERVAL AND NEXT NEAREST INTERVAL AND ROUND UP TO NEAREST INTERVAL
510
GENERATE INPUT STRINGS TO HASH ALGORITHM
514
APPLY HASH ALGORITHM AND GENERATE TICKETS
518
DOES A STREAMING SERVER TICKET MATCH WEB SERVER TICKET?
522
YES
HAS END USER ALREADY VIEWED STREAM?
530
NO
YES
NO
STORE RECEIVED END USER ID AND STREAM ID
534
PROVIDE ACCESS TO MEDIA CONTENT
538
DENY ACCESS TO MEDIA CONTENT
526

FIG. 5

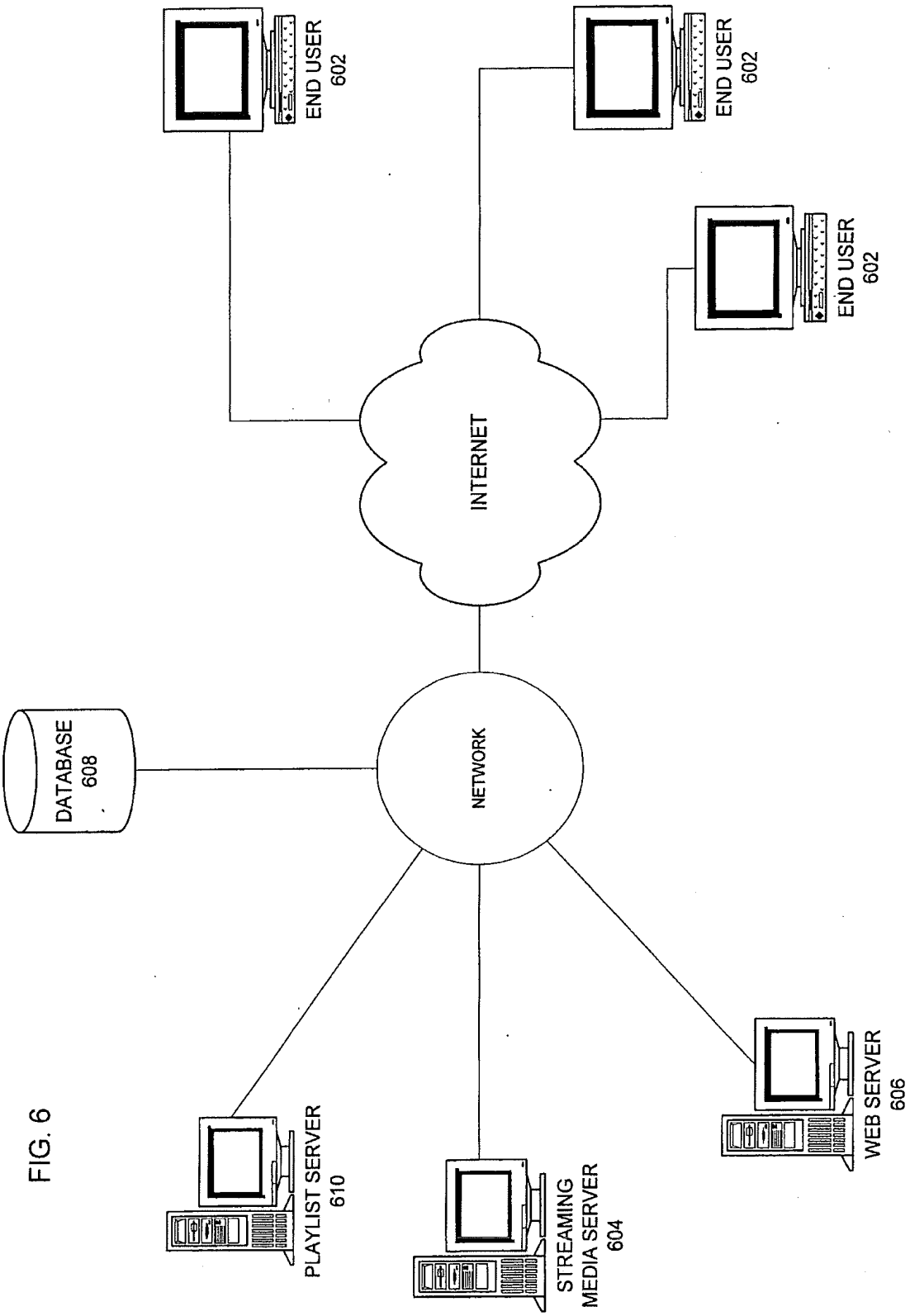

END USER
602
END USER
602
END USER
602
INTERNET
DATABASE
608
NETWORK
PLAYLIST SERVER
610
STREAMING
MEDIA SERVER
604
WEB SERVER
606

FIG. 6

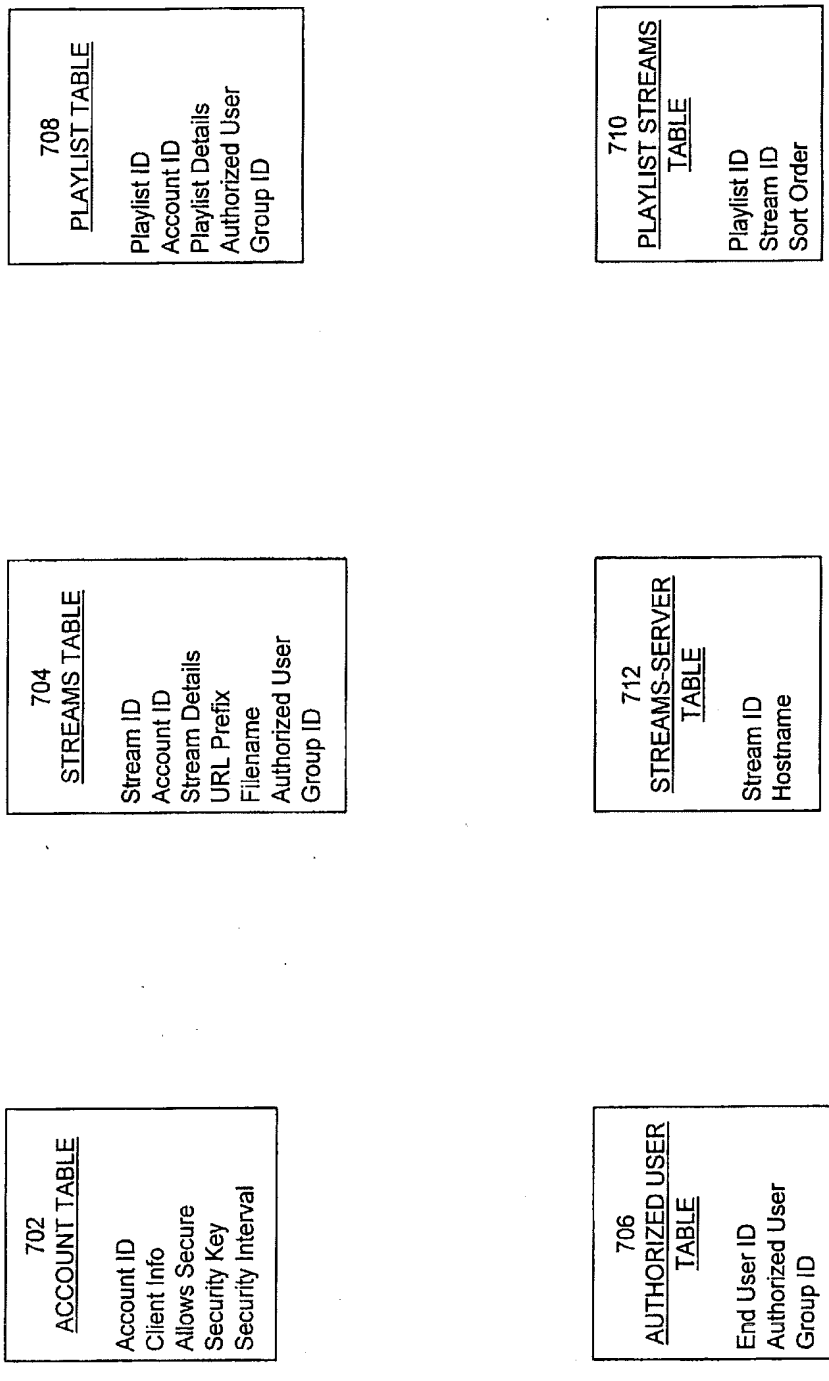
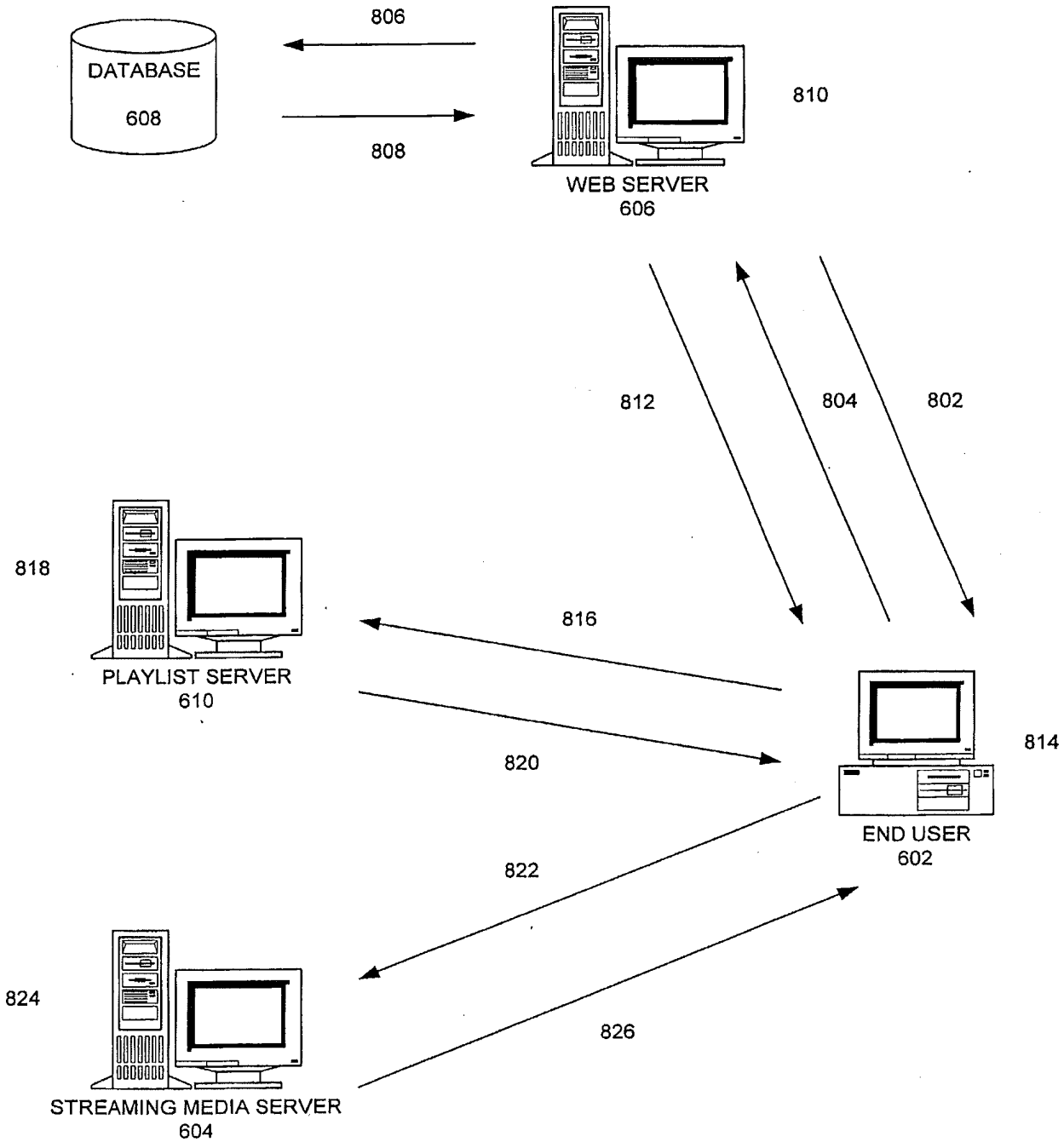

702
ACCOUNT TABLE
Account ID
Client Info
Allows Secure
Security Key
Security Interval
704
STREAMS TABLE
Stream ID
Account ID
Stream Details
URL Prefix
Filename
Authorized User
Group ID
706
AUTHORIZED USER TABLE
End User ID
Authorized User
Group ID
708
PLAYLIST TABLE
Playlist ID
Account ID
Playlist Details
Authorized User
Group ID
710
PLAYLIST STREAMS TABLE
Playlist ID
Stream ID
Sort Order
712
STREAMS-SERVER TABLE
Stream ID
Hostname

FIG. 7

FIG. 8


DATABASE
608
806
808
WEB SERVER
606
810
812
804
802
818
PLAYLIST SERVER
610
816
820
814
END USER
602
822
824
826
STREAMING MEDIA SERVER
604

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US01/18324

A. CLASSIFICATION OF SUBJECT MATTER

IPC(7) : GO6F 11/00, 1/24
US CL : 713/200, 201, 202, 100.
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system followed by classification symbols)

U.S. : 713/200, 201, 202, 100.

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)

East-DERWENT, JPO, EPO, USPAT FULL, PG PUB, IBMTDB. search terms; media, server, ticket, authorization, stream, data, pay, view, play, list.

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 6,032,260 A (SASMAZEL et al.) 29 Feburary 2000 see entire document.	1-21
Y,E	US 6,263,432 B1 (SASMAZEL et al.) 17 July 2001, see entire document.	1-21

☐ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

*	Special categories of cited documents:	"T"	later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"A"	document defining the general state of the art which is not considered to be of particular relevance	"X"	document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"E"	earlier document published on or after the international filing date	"Y"	document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"L"	document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)	"&"	document member of the same patent family
"O"	document referring to an oral disclosure, use, exhibition or other means		
"P"	document published prior to the international filing date but later than the priority date claimed		

Date of the actual completion of the international search 01 OCTOBER 2001	Date of mailing of the international search report 25 OCT 2001
Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231 Facsimile No. (703) 305-3230	Authorized officer NORMAN MICHAEL WRIGHT James R. Matthews Telephone No. (703) 308-0000

Form PCT/ISA/210 (second sheet) (July 1998)★

第 6 部門第 3 区分　　全 17 頁

(19)日本国特許庁(JP)　(12)公表特許公報(A)　(11)特許出願公表番号

特表2004-533690

(P2004-533690A)

(43)公表日　平成16年11月4日(2004.11.4)

(51)Int.Cl.[7]	F I			テーマコード（参考）
G06F 12/14	G06F	12/14	520D	5B017
G06F 12/00	G06F	12/14	520F	5B082
	G06F	12/00	537A	

審査請求 有　予備審査請求 有　（全 72 頁）

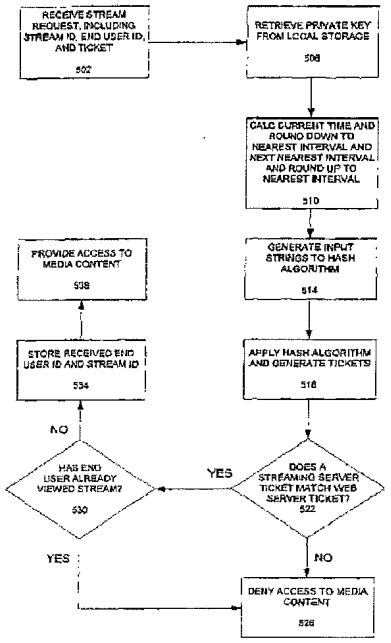
(21)出願番号　特願2003-502687(P2003-502687)
(86)(22)出願日　平成13年6月6日(2001.6.6)
(85)翻訳文提出日　平成15年12月8日(2003.12.8)
(86)国際出願番号　PCT/US2001/018324
(87)国際公開番号　WO2002/099640
(87)国際公開日　平成14年12月12日(2002.12.12)
(81)指定国　AP(GH,GM,KE,LS,MW,MZ,SD,SL,SZ,TZ,UG,ZW),EA(AM,AZ,BY,KG,KZ,MD,RU,TJ,TM),EP(AT,BE,CH,CY,DE,DK,ES,FI,FR,GB,GR,IE,IT,LU,MC,NL,PT,SE,TR),OA(BF,BJ,CF,CG,CI,CM,GA,GN,GW,ML,MR,NE,SN,TD,TG),AE,AG,AL,AM,AT,AU,AZ,BA,BB,BG,BR,BY,BZ,CA,CH,CN,CO,CR,CU,CZ,DE,DK,DM,DZ,EE,ES,FI,GB,GD,GE,GH,GM,HR,HU,ID,IL,IN,IS,JP,KE,KG,KP,KR,KZ,LC,LK,LR,LS,LT,LU,LV,MA,MD,MG,MK,MN,MW,MX,MZ,NO,NZ,PL,PT,RO,RU,SD,SE,SG,SI,SK,SL,TJ,TM,TR,TT,TZ,UA,UG,US,UZ,VN,YU,ZA,ZW

(71)出願人　501438485
ヤフー！　インコーポレイテッド
アメリカ合衆国　カリフォルニア州　９４０８９　サニーヴェイル　ファースト　アヴェニュー　７０１
(74)代理人　100104156
弁理士　龍華　明裕
(72)発明者　マディソン　ジャスティン
アメリカ合衆国、７５０８１　テキサス州、　リチャードソン、　コンコルド　１５０４
(72)発明者　ロディガー　アンソニー
アメリカ合衆国、７５０５６　テキサス州、　ザ　コロニー、　レイク　ヴィスタ　ドライブ　５００４

最終頁に続く

(54)【発明の名称】デジタル・コンテンツおよびストリーミングデータへのアクセスを管理するシステム及び方法

(57)【要約】
【課題】ストリーミング・メディアを含むデジタル・コンテンツへのアクセスを管理するシステムおよび方法。
【解決手段】ストリーミング・データ（５０２）を含むデジタル・コンテンツへのアクセスを管理するシステムおよび方法であって、メディアサーバは認証チケットを生成しそれをウェブサーバ（５１８）により生成されたものと一致するかを比較（５２２）してアクセスを認可するかどうかを判断する（５３０）。



【特許請求の範囲】
【請求項１】
１つ以上のメディア・ファイルへのアクセスを管理する方法であって、
ａ．第１時間に基づいて第１認証チケットを生成するステップと、
ｂ．ストリーミング・リクエストおよび前記認証チケットをサーバへ送信するステップと、
ｃ．第２時間に基づいて第２認証チケットを生成するステップと、
ｄ．前記メディア・ファイルへのアクセスを許可するかどうかの判断を可能にするために、前記第１認証チケットと前記第２認証チケットとを比較するステップと
を備える方法。
【請求項２】
ステップ（ａ）がおおよそ前記第１時間に起こり、ステップ（ｃ）がおおよそ前記第２時間に起こり、前記第２時間は前記第１時間よりも未来の時間であり、
前記第１認証チケットと前記第２認証チケットが一致する場合、前記方法は、前記メディア・ファイルへのアクセスを許可するステップをさらに備え、
前記第１時間と前記第２時間との偏差が所定の量よりも少ない場合に前記第１認証チケットと前記第２認証チケットが一致する、
請求項１に記載の方法。
【請求項３】
ステップ（ａ）がおおよそ前記第１時間に起こり、ステップ（ｃ）がおおよそ前記第２時間に起こり、前記第２時間は前記第１時間よりも未来の時間であり、
前記第１認証チケットと前記第２認証チケットが一致しない場合、前記方法は、前記メディア・ファイルへのアクセスを拒否するステップをさらに備え、
前記第１時間と前記第２時間との偏差が所定の量よりも多い場合に前記第１認証チケットと前記第２認証チケットが一致しない、
請求項１に記載の方法。
【請求項４】
前記第１認証チケットは、前記第１時間を或る時間間隔の倍数まで切り下げた値と近似する第１時刻に更に基づき、前記第２認証チケットは、前記第２時間を前記時間間隔の倍数まで切り下げた値と近似する第２時刻に更に基づく、請求項３に記載の方法。
【請求項５】
前記第１認証チケットおよび前記第２認証チケットが、セキュリティ・キーに更に基づく、請求項１に記載の方法。
【請求項６】
前記第１認証チケットおよび前記第２認証チケットが、前記メディア・ファイルの識別子に更に基づく、請求項１に記載の方法。
【請求項７】
前記ファイルが再生リストを備え、前記識別子が前記再生リストの識別子である、請求項６に記載の方法。
【請求項８】
前記第１認証チケットが、エンド・ユーザからの前記ファイルへのアクセスを求めるリクエストに応答して生成され、前記第１認証チケットおよび前記第２認証チケットが、前記エンド・ユーザの識別子に更に基づく、請求項１に記載の方法。
【請求項９】
第３時間に基づいて第３認証チケットを生成するステップを更に備え、
前記メディア・ファイルへのアクセスを許可するかどうかを判断するために、ステップ（ｄ）が、前記第１認証チケットから前記第３認証チケットを比較するステップを更に備える、請求項１に記載の方法。
【請求項１０】
前記第２認証チケットを生成するステップがおおよそ前記第２時間に起こり、前記第２認証チケットは、前記第２時間を或る時間間隔の他の倍数まで切り下げた値と近似する第２時刻に基づき、
前記第３認証チケットを生成するステップがおおよそ前記第３時間に起こり、前記第３認証チケットは、前記第３時間を前記時間間隔の倍数まで切り下げた値と近似する第３時刻に基づき、前記第３時刻は前記第２時刻よりも過去の時刻である、請求項９に記載の方法。
【請求項１１】
前記第２時間と前記第３時間が同一の時間である、請求項１０に記載の方法。
【請求項１２】
前記第２認証チケットを生成するステップがおおよそ前記第２時間に起こり、前記第２認証は、前記第２時間を或る時間間隔の倍数まで切り下げた値と近似する第２時刻に更に基づき、
前記第３認証チケットを生成するステップがおおよそ前記第３時間に起こり、前記第３認証チケットは、前記第３時間を前記時間間隔の他の倍数まで切り上げた値と近似する第３時刻に更に基づく、請求項９に記載の方法。
【請求項１３】
前記第２時間と前記第３時間が同一の時間である、請求項１２に記載の方法。
【請求項１４】
１つ以上のメディア・ファイルへのアクセスを管理する方法であって、
ａ．前記メディア・ファイルに対するリクエストを受信するステップと、
ｂ．前記リクエストに関連する第１認証チケットを受信するステップと、
ｃ．第２時間に基づいて第２認証チケットを生成するステップと、
ｄ．前記第１認証チケットと前記第２認証チケットとを比較する事によって、前記メディア・ファイルへのアクセスを許可するかどうかを判断するステップと
を備え、
前記第１認証チケットが第１時間に基づく、
方法。
【請求項１５】
ｅ．前記第１認証チケットと前記第２認証チケットが一致しないと言う事に基づいて、前記メディア・ファイルへのアクセスを拒否するステップをさらに備え、
前記第１時間と前記第２時間との偏差が所定の量である場合に前記第１認証チケットと前記第２認証チケットが一致しない、
請求項１４に記載の方法。
【請求項１６】
前記第１認証チケットは、前記第１時間を或る時間間隔の倍数まで切り下げた値と近似する第１時刻に更に基づき、前記第２認証チケットは、前記第２時間を前記時間間隔の前記倍数まで切り下げた値と近似する第２時刻に更に基づく、請求項１４に記載の方法。
【請求項１７】
前記第１認証チケットおよび前記第２認証チケットが、セキュリティ・キーに更に基づく、請求項１４に記載の方法。
【請求項１８】
前記第１認証チケットおよび前記第２認証チケットが、前記メディア・ファイルの識別子に更に基づく、請求項１４に記載の方法。
【請求項１９】
前記ファイルが再生リストを備え、前記識別子が前記再生リストの識別子である、請求項１８に記載の方法。
【請求項２０】
前記第１認証チケットが、エンド・ユーザからの前記ファイルへのアクセスを求めるリクエストに応答して生成され、前記第１認証チケットおよび前記第２認証チケットが、前記エンド・ユーザの識別子に更に基づく、請求項１４に記載の方法。
【請求項２１】
第３時間に基づいて第３認証チケットを生成するステップを更に備え、
前記メディア・ファイルへのアクセスを許可するかどうかを判断するために、ステップ（ｄ）が、前記第１認証チケットから前記第３認証チケットを比較するステップを更に備える、請求項１４に記載の方法。
【請求項２２】
前記リクエストがメディア・プレーヤーから受信される、請求項１４に記載の方法。
【請求項２３】
前記第１認証チケットが、ウェブ・サーバによって生成される、請求項１４に記載の方法。

【請求項２４】
ステップ（ａ）からステップ（ｄ）までがメディア・サーバによって実行される、請求項１４に記載の方法。
【請求項２５】
１以上のメディア・ファイルへのアクセスをエンド・ユーザが受信する方法であって、
ａ．前記メディア・ファイルへのアクセスをリクエストするステップと、
ｂ．第１認証チケットをおおよそ第１時間に生成させるステップと、
ｃ．第２認証チケットをおおよそ第２時間に生成させるステップと、
ｄ．前記第１認証チケットと前記第２認証チケットが一致する場合、前記メディア・ファイルへのアクセスを受信するステップと
を備え、
前記第１認証チケットは前記第１時間に基づき、
前記第２認証チケットは前記第２時間に基づき、
前記第１時間と前記第２時間との偏差が所定の量よりも少ない場合に前記第１認証チケットと前記第２認証チケットが一致する、
方法。
【請求項２６】
ｅ．前記エンド・ユーザの識別子を供給するステップを更に備え、前記第１認証チケットと前記第２認証チケットが前記識別子に更に基づく、請求項２５に記載の方法。
【請求項２７】
ステップ（ｂ）が、ウェブ・ページ上のリンクを起動するステップを備える、請求項２５に記載の方法。
【請求項２８】
ステップ（ｃ）が、メディア・サーバを呼び出すステップを備える、請求項２５に記載の方法。
【請求項２９】
ステップ（ｄ）が、メディア・プレーヤーによって前記メディア・ファイルを受信するステップを更に備える、請求項２５に記載の方法。
【請求項３０】
１つ以上のメディア・ファイルへのアクセスを管理するシステムであって、
ソフトウエアと共働して、第１時間に基づいて第１認証チケットを生成する第１プロセッサと、
前記第１プロセッサからは独立しており、ソフトウエアと共働して、第２時間に基づいて第２チケットを生成する第２プロセッサと、
ソフトウエアと共働して、前記第１認証チケットを受信し、前記第１認証チケットと前記第２認証チケットとを比較する事によって、前記メディア・ファイルへのアクセスを許可するかどうかを判断する第３プロセッサと
を備えるシステム。
【請求項３１】
前記第２プロセッサが前記第３プロセッサである、請求項３０に記載のシステム。
【請求項３２】
前記第１プロセッサが更に、第１時刻に基づいて前記第１認証チケットを生成し、前記第１時間を或る時間間隔の倍数まで切り下げることによって前記第１時刻を計算し、前記第２プロセッサが更に、第２時刻に基づいて前記第２認証チケットを生成し、前記第２時間を前記時間間隔の前記倍数まで切り下げることによって前記第２時刻を計算する、請求項３０に記載のシステム。
【請求項３３】
前記第２プロセッサが更に、第３時間に基づいて第３認証チケットを生成し、前記第３プロセッサが更に、第１認証チケットと第３認証チケットを比較する、請求項３０に記載のシステム。
【請求項３４】
前記第２時間と前記第３時間が同一の時間である、請求項３３に記載のシステム。
【請求項３５】
前記第２プロセッサが更に、第３時刻に基づいて前記第３認証チケットを生成し、前記第３時間を或る時間間隔の倍数まで切り上げることによって前記第３時刻を計算する、請求項３３に記載のシステム。
【請求項３６】
前記第１プロセッサおよび前記第２プロセッサが、前記メディア・ファイルの識別子に基づいて前記第１および第２認証チケットをそれぞれ生成する、請求項３０に記載のシステム。
【請求項３７】
前記第１プロセッサおよび前記第２プロセッサが、セキュリティ・キーに基づいて前記第１および第２認証チケットをそれぞれ生成する、請求項３０に記載のシステム。
【請求項３８】
前記第１プロセッサおよび前記第２プロセッサが、前記メディア・ファイルへのアクセスをリクエストしているエンド・ユーザの識別子に基づいて前記第１および第２認証チケットをそれぞれ生成する、請求項３０に記載のシステム。
【請求項３９】
コンピュータ・プログラムを備えるコンピュータ可読媒体であって、前記コンピュータ・プログラムが１つ以上のプロセッサに対し、
ａ．メディア・ファイルと関連付けられている第１認証チケットを受信し、
ｂ．第２時間に基づいて第２認証チケットを生成し、
ｃ．前記第１認証チケットと前記第２認証チケットとを比較し、
ｄ．前記第１認証チケットと前記第２認証チケットとの比較に基づいて、前記メディア・ファイルが送信されるようにする
ように命令し、
前記第１認証チケットが第１時間に基づく、
コンピュータ可読媒体。

【発明の詳細な説明】
【技術分野】
【０００１】
本発明は、広義にはデジタル・コンテンツへのアクセスを管理する事に関し、より詳細には、時間に一部基づくチケットに基づいてシステムおよびストリーミング・メディアへのアクセスを制限する方法に関する。
【背景技術】
【０００２】
インターネットとワールドワイド・ウェブの普及により、ストリーミング・メディア・コンテンツのようなデジタル・コンテンツの配信に関連する産業が発展した。例えば、ストリーミング・メディアは、エンターテインメント、通信教育および企業目的を含む多数の目的のうちの何れかのために使用され得る。エンターテインメント会社は映画とスポーツイベントをストリーミングし、通信教育会社は教育のコンテンツをストリーミングし、また、企業はトレーニング教材をストリーミングする。
【発明の開示】
【発明が解決しようとする課題】
【０００３】
このようなストリーミング・メディアの多くの用途において、コンテンツへのアクセスを管理する事は必要不可欠である。例えば、エンターテインメント会社は、ストリーミング・メディアのアイテムをエンド・ユーザが視聴する毎に課金し得るが、これはエンターテインメント業界においては「ペイ・パー・ビュー」と称されている。同様に、通信教育会社は、オンライン教育のコースへのアクセス、およびストリーミング・メディアへのアクセスに対して、学生に課金する。企業関係のコンテンツは多くの場合機密であり、従って、管理されたアクセスを必要とする。
【０００４】
従って、ストリーミング・メディアへのアクセスを制限するために各種のシステムが開発されている。ストリーミング・コンテンツへのアクセスを制限するため

の現在の業界標準は、ストリーミング・メディア・コンテンツを提供するのに先立ってエンド・ユーザを確認するストリーミング・メディア・サーバを含んでいる。より具体的には、ストリーミング・メディア・サーバは、ストリーミング・メディアへのアクセスを許可するかどうかを判断するロジックを含むコンパイルされたプログラムのソフトウェア・プラグインを含んでいる。しかしながら、そのような認証プラグインはしばしば複雑になり、開発・維持が困難である。例えば、ストリーミング・メディア・コンテンツへのアクセスを許可するロジックを変更する必要性が生じた場合、ストリーミング・メディア・サーバ上のコンパイルされたプラグインの変更は困難である。更に、ストリーミング・メディア・サーバに存在するロジックの全てにおいて、ストリーミング・メディア・サーバは、データベースあるいは各所に配置されるメッセージ転送サービスに対して直接アクセスしていなければならない。さらに、ストリーミング・メディア・コンテンツにアクセスすることを認められているような特定のエンド・ユーザを確認するような例においてさえも、そのようなエンド・ユーザは多くの場合無許可のエンド・ユーザとアクセスを共有することにより許可プロセスの裏をかくことができる。このようなアクセスの共有は、コンテンツへのリンクのユーザ名およびパスワードを共有する事のような多くの形式を取り得る。同様の問題が、他の形式のデジタル・コンテンツへのアクセスを管理するために使用されるシステムにおいても存在する。従って、デジタル・コンテンツ、具体的にはストリーミング・メディア・コンテンツへのアクセスを管理し、エンド・ユーザを許可するための改善されたシステムおよび方法が必要となっている。

【課題を解決するための手段】

【０００５】

本発明は、オーディオ、ビジュアル、ビデオ、テキスト、ストリーミング・メディアのようなデジタル・コンテンツへのアクセスを管理するシステムおよび方法を提供する事によって、上記の、および他のニーズを解決する。本発明に係る１つのシステムおよび方法はストリーミング・メディアへのアクセスを管理し、ネットワークに接続されたウェブ・サーバ、メディア・サーバ、およびパーソナルコンピュータのようなエンド・ユーザ・プロセッサを備える。

に、本願明細書において記載される実施形態はオンデマンドのストリーミング・メディアに関するものであるが、本実施例がライブのストリーミング・メディアに適用できることは、当業者にとって認識される。

【００１０】

一般に、本実施形態のシステムは、エンド・ユーザ・プロセッサ１０２、ストリーミング・メディア・サーバ１０４、およびコンテンツ・マネジメント（ＣＭ）データベース１０８を備えるウェブ・サーバ１０６を有し、これらの全てがインターネットに接続される。エンド・ユーザ・プロセッサ１０２は、インターネット・エクスプローラーと言う名称でマイクロソフト社より提供されるか、または、ネットスケープ・ナビゲータと言う名称でネットスケープ・コミュニケイションズより提供されるようなインターネット・ブラウザ、および、ウインドウズ・メディア・プレーヤーと言う名称でマイクロソフト社より提供される、また、リアルプレーヤーと言う名称でリアル・ネットワークス社によって提供されるストリーミング・メディア・プレーヤーを含む。ウェブ・サーバ１０６はエンド・ユーザ１０２によってアクセス可能なウェブサイトを提供する。ストリーミング・メディア・サーバ１０４上に存在するストリーミング・メディア・コンテンツにアクセスするために、ウェブサイトはエンド・ユーザ１０２によって起動可能なリンクを含む。

【００１１】

本発明は、任意のコンピュータ技術の使用に際して適用し得る事が理解されよう。例えば、本実施形態はインターネットによってコンテンツへのアクセスを提供する事に関するが、本発明は、例えば広域ネットワークを含む任意のコンピュータネットワーク上に使用され得る。同様に、エンド・ユーザ・プロセッサ１０２は、例えばＰＤＡ、ウェブ対応の携帯電話、ネットワークにダイアルアップする有線電話、モバイル・コンピュータ、パーソナルコンピュータ、インターネット装置等のネットワークに接続し得る任意のデバイスであり得る。更に、ここに記述されたサーバは、任意のソフトウエアを実行する任意の種類のもので有り得、また、ここに記述されたソフトウェア・モジュール、オブジェクトおよびプラグインは、任意のプログラム言語で記述され得る。最後に、ここに記述されたデータベースおよび記憶装置は、例えばローカルのコンピュータ・メモリー、ネットワークに接続された記憶装置、および磁気あるいは光学のような任意の公知の記憶媒体の様な任意の記憶技術を利用し得る。

【００１２】

ＣＭデータベース１０８の一例が図２に示される。図示されるように、データベース１０８は、全てのストリーミング・コンテンツに普遍的に適用可能な情報と、関連データの幾つかのテーブルを含む。普遍的な情報２０２はセキュリティ・キー、セキュリティ・インターバル、およびコンテンツが保存されているストリーミング・メディア・サーバ１０４の名前（「ホスト名」）を含んでいる。セキュリティ・キーおよびセキュリティ・インターバルは、エンド・ユーザ１０２がコンテンツにアクセスすることを許可するために使用され、秘密にされ、かつ、コンテンツの所有者によって設定される事が望ましい。セキュリティ・キーおよびセキュリティ・インターバルは全てのコンテンツに対するアクセスを管理するために使用されるが、他の実施形態においては、各コンテンツ・ファイルがそれ自身に関するセキュリティ・キーおよびセキュリティ・インターバルを有する。

【００１３】

ＣＭデータベース１０８は、コンテンツあるいはストリーミング識別情報を含む一連のテーブルをさらに有する。より詳細には、ストリーミング・テーブル２０４は、ユニークなストリーミング識別子（ＩＤ）によって識別されるそれぞれのストリーミング・コンテンツのレコードを含む。更に、各レコードは、例えばコンテンツ・ファイルの作成日を含むコンテンツ・ファイルについて記述するストリーミングの詳細、ファイルの記述、コンテンツがオーディオであるかビデオであるかの識別子、コンテンツが関係するプラットフォーム、コンテンツが最後に修正された日付、コンテンツを視聴するのに必要なコーデック、コンテンツの長さおよびサイズ、コンテンツの有効期限（もしあれば）、．ａｓｆまたは．ｒｍのようなストリーミングの種類、コンテンツのタイトルおよびコンテンツの作者、コンテンツのステータス、コンテンツの著作権情報、コンテンツのビットレート等を含む。各レコードは更に、メディア・サーバ１０４へのリンクを生成す

【０００６】

運用において、エンド・ユーザのファイルのアクセスに対するリクエストに応答して、暗号化されたチケットを生成する。チケットは、チケットが生成された時間、またはそれに近接した時間に少なくとも一部基づく。一実施形態において、チケットは、例えばセキュリティ・タイム・インターバルまたはエンド・ユーザ識別子のような付加的な情報に基づく。

【０００７】

メディア・サーバがリクエストされたファイルへのアクセスを提供するのに先立って、メディア・サーバは、好ましくはウェブ・サーバと同一の暗号化アルゴリズムを用いて認証チケットを生成する。メディア・サーバ認証チケットは、メディア・サーバがファイルへのアクセスに対するリクエストを受信した時間、またはそれに近接した時間に少なくとも一部基づく。メディア・サーバは、ウェブ・サーバによって生成されたチケットとメディア・サーバによって生成されたチケットを比較する事によって、ファイルへのアクセスを許可するかどうかを判断する。

【０００８】

一実施形態において、これらのチケットが一致しない場合、ウェブ・サーバがチケットを生成した時間とメディア・サーバがチケットを生成した時間との差が、所定の量よりも大きく、チケットは論理的に「有効期限切れ」であると考えられる。従って、メディア・サーバは、メディア・コンテンツに対するアクセスを許可しない。これらのチケットが一致する場合、これらのチケットは認可されたタイム・インターバルの範囲内に生成されており、メディア・サーバは、エンド・ユーザがリクエストしたメディア・コンテンツにアクセスする事を許可する。

【発明を実施するための最良の形態】

【０００９】

本発明の特定の好ましい実施形態は、図面に関して記載されている。コンテンツに対するアクセスを管理するための本発明は、ストリーミング・メディア・ファイルに対するアクセスの管理に関するコンテキストで記載されているが、本発明が全ての種類のメディアまたはファイルに適用され得る事が理解されよう。さら

るために使用される接頭部　（「ＵＲＬ接頭部」）、およびストリーミング・メディア・サーバ１０４に格納されているコンテンツ・ファイルの名前（「ファイル名」）を含む。ファイル名は、オンデマンドのコンテンツ用のストリーミング・メディア・サーバ１０４に接続された記憶装置上の実際のパスを指しても良いし、ファイル名は、ライブのストリーミング用の別名、チャネルまたはポートを指しても良い、と言う事が理解されるべきである。

【００１４】

データベース１０８は「再生リスト」情報を有するテーブルを含む。クライアントの再生リストは、一般に、グループとして利用可能になされる目的で論理上関連した、１以上のコンテンツ・ファイルのグループである。再生リストの一部であると確認された各コンテンツ・ファイルもまた個々に利用可能になされ得る。このような再生リスト情報は、再生リストテーブル２０８および再生リスト・ストリーミング・テーブル２１０に含まれる。一般に、再生リストテーブル２０８は、再生リストＩＤによって識別される各再生リストを識別するレコードを含む。各レコードは、例えば再生リストのフォーマット（例えばウインドウズ・メディア・プレーヤーまたはリアルプレーヤー）、再生リストの記述、再生リスト名および同種のもの、および再生リストに対する許可ユーザ・グループＩＤを含む、再生リスト詳細をさらに含む。

【００１５】

許可ユーザ・グループＩＤは、特定の再生リストを視聴することを認められるエンド・ユーザ１０２のグループに対応する。より詳細には、データベース１０８は、ユニークなエンド・ユーザＩＤによって識別される各エンド・ユーザ１０２を１以上の許可ユーザ・グループＩＤに関連付ける許可ユーザ・テーブル２０６をさらに含む。エンド・ユーザ１０２が再生リストを視聴するために、エンド・ユーザ１０２が該コンテンツ・ファイルのための許可ユーザ・グループＩＤの一部であるということが確認されなければならない。他の実施形態においては許可グループＩＤは使用されない場合もあり、更に他の実施形態においては、コンテンツ・ファイルのそれぞれが、該ファイルに関連付けられた許可グループＩＤを有する。

【００１６】

再生リスト・ストリーミング・テーブル２１０は、再生リストＩＤによって識別される再生リストと、ストリーミングＩＤによって識別される構成コンテンツ・ファイルとを関連付けるレコードを有する。さらに、各レコードは、再生リスト中の各コンテンツ・ファイルの順番を示す情報（「ソート順番」）を含んでいる。

【００１７】

本実施形態において使用されるコンポーネントについて記載してきたが、ここで、ストリーミング・メディア・コンテンツへのアクセスを管理するプロセスについて記述する。概説すると、ウェブ・サーバ１０６に保存された許可ソフトウエア・コンポーネントは、公開暗号キー情報、プライベート・キー情報および現在時刻に基づいたハッシュ値即ち「チケット」を生成する。公開暗号キーは、エンド・ユーザ１０２およびエンド・ユーザのユーザＩＤによってリクエストされるストリーミング・コンテンツが有するユニーク識別子である。プライベート・キーは、コンテンツの所有者によって設定されたセキュリティ・キーとセキュリティ・インターバルを含む。

【００１８】

リクエストされたコンテンツが格納されているストリーミング・メディア・サーバ１０４は、公開暗号キーを含むストリーミング・リクエストと、ウェブ・サーバ１０６によって生成されたチケットを受信する。ストリーミング・メディア・サーバ１０４は、自身のチケットを生成するために、ローカルに格納されたプライベート・キー情報を使用するプロセスへと進む。ストリーミング・メディア・サーバ１０４は、ストリーミング・メディア・サーバ１０４およびウェブ・サーバ１０６によって生成されるチケットを比較することによって、リクエストされたストリーミング・メディアを供給するか、もしくは供給を拒否する。

【００１９】

アクセスを管理するプロセスが、図３のワークフロー・ダイアグラムおよび図４および図５のフローチャートによって詳細に記述される。本実施例において、エンド・ユーザ１０２があるストリーミング・メディア・コンテンツ・ファイルへのアクセスをリクエストする。最初、ウェブ・サーバ１０６は、認証アプリケーションにログインすることをエンド・ユーザにリクエストし、あるストリーミング・メディアを視聴する選択肢をエンド・ユーザに提供するウェブ・ページを供する。（ステップ３０２）。例えば、このようなページは、いくつかのコンテンツ・ファイルのうち、視聴するための１つのコンテンツを選択することをエンド・ユーザにリクエストするフォームを含んでいてもよく、このフォームは、コンテンツの所有者が前もって割り当ててエンド・ユーザに提供したエンド・ユーザＩＤと、選択するコンテンツのアクセスに対してエンド・ユーザが課金されるようにクレジットカード番号を提供するために用いられる。他の実施形態において、エンド・ユーザはエンド・ユーザの連絡情報および請求書発行情報を提供することによりコンテンツの所有者にあらかじめ登録され、コンテンツの所有者は、エンド・ユーザの連絡情報および請求書発行情報を、登録されたエンド・ユーザＩＤと共にテーブル形式で保存する。

【００２０】

ウェブ・ページに応答して、エンド・ユーザはエンド・ユーザのユーザＩＤを提供し、リンクを起動し、これによって、認証アプリケーションにログインし、該リンクに関連する特定のストリーミング・メディア・コンテンツ・ファイルへのアクセスをリクエストする。（ステップ３０４）。ストリーミングＩＤが“１２３４５６”で表されるストリーミング・リクエストの一例が以下に示される。＜Ａ　ｈｒｅｆ　ｈｔｔｐ：／／ｗｅｂｓｅｒｖｅｒ．ｃｏｍｐａｎｙ．ｃｏｍ／ｇｅｔｓｔｒｅａｍ．ａｓｐ？ＩＤ＝１２３４５６＞。

【００２１】

本実施形態において、認証アプリケーションは、ウェブ・サーバ１０６上に存在する「．ｄｌｌ」ソフトウエア・コンポーネントである。しかしながら、ここに記述された機能性を実施するために、例えばアクティブ・サーバー・ページ（ＡＳＰ）またはサーブレットのような他のプログラム言語あるいは技術を使用し得ることを、当業者は認識するであろう。特定のプログラミング技術に関係なく、認証アプリケーションは、エンド・ユーザ・プロセッサ１０２上の如何なる処理障害をも緩和するためにウェブ・サーバ１０６上で実行されることが望ましい。

【００２２】

一旦エンド・ユーザが認証アプリケーションにログインし、ウェブ・サーバ１０６がエンド・ユーザからストリーミング・リクエストおよびエンド・ユーザＩＤを受け取れば、ウェブ・サーバ１０６は、認証チケットを動的に生成し、選択されたコンテンツ・ファイルへのリンクを動的に生成することを継続する。より詳細には、認証アプリケーションの管理下において、ウェブ・サーバ１０６は、認証チケットを生成する際に使用するためのプライベート・キーをデータベース１０８に対してリクエストする。（ステップ３０６）。ウェブ・サーバ１０６は、リクエストされたコンテンツ・ファイルに関連したセキュリティ・キーおよびセキュリティ・インターバルを含むプライベート・キーをＣＭデータベース１０８から検索するためにデータベース・クエリーを発行する。これを受けて、ＣＭデータベース１０８はウェブ・サーバ１０６にプライベート・キーを返送する。（ステップ３０８）。

【００２３】

データベース１０８からプライベート・キーを取得した後、ウェブ・サーバ１０６はチケットを生成する。（ステップ３１０）。図４に関してより詳細に記述されるように、ウェブ・サーバ１０６は、チケットを生成するためにプライベート・キー、ストリーミングＩＤ、エンド・ユーザＩＤ、現在時刻およびハッシュ・アルゴリズムを利用する。本実施形態において、リクエストされたコンテンツのストリーミングＩＤが、ステップ３０４においてエンド・ユーザによって起動されたストリーミング・リクエスト・リンクに含まれているので、ウェブ・サーバ１０６は、チケットを生成する際にストリーミングＩＤを使用することができる。しかしながら、他の実施形態においては、エンド・ユーザによって提供されるストリーミング・リクエストは、例えばコンテンツのタイトル、作者、および（または）ファイル名の様な、ストリーミングＩＤ以外のユニーク識別情報を含む。このような実施形態において、ウェブ・サーバ１０６はストリーミング・テーブル２０４を検索し、ストリーミング・リクエストに含まれる識別情報に基づいてストリーミングＩＤを検索する。更に他の実施形態において、ストリーミング・リクエストは、チケットがシステムを生成する際に使用するファイル名また

はパスのような、ストリーミングＩＤ以外のユニーク識別子を含む。

【００２４】

一旦チケットが生成されれば、ウェブ・サーバ１０６はメディア・サーバ１０４上に存在するリクエストされたコンテンツへのリンクを生成する。より詳細には、上に例示したストリーミング・リクエストに基づいて、エンド・ユーザ・プロセッサ１０２に存在するメディア・プレーヤーが、メディア・サーバ１０４へのリンクを動的に生成するためのプログラム「ｇｅｔｓｔｒｅａｍ．ａｓｐ」を実行する「ｗｅｂｓｅｒｖｅｒ．ｃｏｍｐａｎｙ．ｃｏｍ」（即ち、ウェブ・サーバ１０６）へ回線を接続する。（ステップ３１２）。当業者は、「ｇｅｔｓｔｒｅａｍ」アプリケーションはアクティブ・サーバー・ページ（即ちＡＳＰ）の拡張子を有しているが、ＡＳＰ技術を使用することが必ずしも必要ではないことを認識するであろう。むしろ、「．ｄｌｌ」コンポーネントの様な任意のプログラミングまたはスクリプト言語または技術が所望の機能性を提供するために使用され得る。しかしながら、認証アプリケーションにおいて、エンド・ユーザ・プロセッサ１０２における如何なる処理障害をも緩和するために、該プログラムはサーバ側で実行されることが望ましい。「ｇｅｔｓｔｒｅａｍ．ａｓｐ」プログラムは、メディア・サーバ１０４へのリンクを動的に生成するのに必要となるデータを検索するため、ウェブ・サーバ１０６がＣＭデータベース１０８を呼び出す様に機能する。より詳細には、ウェブ・サーバ１０６は、普遍情報テーブル２２０およびＵＲＬ接頭部からホスト名を検索し、ストリーミング・テーブル２０４からファイル名を検索する。更に、「ｇｅｔｓｔｒｅａｍ．ａｓｐ」プログラムはストリーミングＩＤ、チケットおよびエンド・ユーザＩＤをリンクの端部に付加する。その後、ウェブ・サーバ１０６は、エンド・ユーザ・プロセッサ１０２のメディア・プレーヤーへリンクを返送する。（ステップ３１４）。

【００２５】

メディア・ファイルへのリンクの実例は以下の通りである。＜ＲＥＦ　ｈｒｅｆ＝”ｍｍｓ：／／ｍｅｄｉａｓｅｒｖｅｒ．ｃｏｍｐａｎｙ．ｃｏｍ／ｓｔｒｅａｍ１．ａｓｆ？ＩＤ＝１２３４５６＆ＴＩＣＫＥＴ＝ｕｖｗ１２３ｘｙｚ＆ＵＳＥＲ＿ＩＤ＝ａｂｃ１２３ｄｅｆ”＞。ここで、ＵＲＬ接頭部は「ｍｍｓ：／／」によってリクエストされ、ホスト名は「ｍｅｄｉａｓｅｒｖｅｒ．ｃｏｍｐａｎｙ．ｃｏｍ」によって表わされ、ファイル名は「ｓｔｒｅａｍ１．ａｓｆ」によって表わされ、リクエストされたコンテンツのストリーミングＩＤは「１２３４５６」によって表わされ、チケットは「ｕｖｗ１２３ｘｙｚ」によって表わされ、そして、エンド・ユーザＩＤは「ａｂｃ１２３ｄｅｆ」によって表わされる。

【００２６】

リンクを受信した後、エンド・ユーザ・プロセッサ１０２はストリーミング・メディア・コンテンツをリクエストするステップに進む。（ステップ３１６）。より詳細には、エンド・ユーザ・プロセッサ１０２に存在するメディア・プレーヤーがリンクによって識別される「ｍｅｄｉａｓｅｒｖｅｒ．ｃｏｍｐａｎｙ．ｃｏｍ」（即ち、ストリーミング・メディア・サーバ１０４）を呼び出す。呼び出しの一部として、メディア・プレーヤーは、リクエストしたコンテンツのストリーミングＩＤのコピー、ウェブ・サーバ１０６によって生成されたチケット、およびエンド・ユーザＩＤをストリーミング・メディア・サーバに１０４に供給する。

【００２７】

ストリーミングＩＤ、エンド・ユーザＩＤおよびチケットを含んでいるリンクを受信した後、ストリーミング・メディア・サーバ１０４は、リクエストされたコンテンツへのアクセスをエンド・ユーザに許可するかどうかを判断するステップに進む。（ステップ３１８）。図５に関してより詳細に以下記述されるように、ストリーミング・メディア・サーバ１０４は、ローカルに格納されたプライベート・キー情報およびリンク中に含まれるストリーミングＩＤおよびエンド・ユーザＩＤに基づいてチケットを個々に生成することにより、アクセスを付与するかどうかを判断する。一般的に、ストリーミング・メディア・サーバ１０４によって生成されたチケットがウェブ・サーバ１０６によって生成されたチケットと一致する場合、ストリーミング・メディア・サーバ１０４はエンド・ユーザ・プロセッサ１０２にリクエストされたストリーミング・メディア・コンテンツを供給する。（ステップ３２０）。

【００２８】

ウェブ・サーバ１０６によってチケットを生成するプロセスが、図４を参照して詳細に記述される。上述したように、チケット生成プロセスは、ウェブ・サーバ１０６に存在する許可ソフトウェア・プラグインによって行なわれることが好ましい。本実施形態において、本プロセスは、ウェブ・サーバ１０６がストリーミングＩＤおよびエンド・ユーザＩＤを含むストリーミング・リクエストを受信するステップより開始される。（ステップ４０２）。その後、ウェブ・サーバ１０６は、リクエストされたストリーミングＩＤに関連したプライベート・キー情報を検索するためにデータベース１０８にアクセスするステップへと進む。（ステップ４０６）。このようなプライベート・キー情報は普遍的なセキュリティ・キーおよびセキュリティ・インターバルを含む。他の実施形態において、各ストリーミングはストリーミング・テーブル２０４にフィールドとしてそれ自身のセキュリティ・キーおよびセキュリティ・インターバルを格納し、ウェブ・サーバ１０６はストリーミング・リクエストに含まれるストリーミングＩＤに基づいてストリーミング・テーブル２０４検索する。

【００２９】

上述されるように、ウェブ・サーバ１０６は、チケットを生成するために現在時刻を更に使用する。より詳細には、ウェブ・サーバ１０６は現在時刻を計算し、セキュリティ・インターバルに最も近い倍数までその時刻を切り下げる。（ステップ４１０）。本実施形態において、Ｃプログラミング言語の標準ライブラリー関数「ｔｉｍｅ（）」によって生成されるような協定世界時（ＵＴＣ）を秒単位で利用する。変数「＄ｔｉｍｅ」によって表される、セキュリティ・インターバルに最も近い倍数まで切り下げられた時間を生成するためのＰｅｒｌプログラム・コードの一例を以下に示す。

【表１】

```
#
# example of 15 minute ticket expiration/security interval
#
$interval = 15 * 60
$time = int (time( ) / $interval) * $interval;
```

【００３０】

ここで、変数「＄ｉｎｔｅｒｖａｌ」はセキュリティ・インターバルに対応しており、セキュリティ・インターバルは１５分である。

【００３１】

例えば、現在の時刻が２０００年５月３１日午後２：１６：０７（中部標準時）であった場合、関数「ｔｉｍｅ（）」は、おおよそ「９５９８００５６７」の値を返す。直近の１５分インターバルにこのＵＴＣ時間を切り下げることによって「９５９８００５００」が得られ、これは、２０００年５月３１日午後２：１５：００（中部標準時）に相当する。

【００３２】

先に例示したコードは変更されても良く、その変更されたコードもまた本発明の範囲内であり得る事が理解されよう。例えば、セキュリティ・インターバルは数分である必要は無く、インターバルが「ｔｉｍｅ（）」関数によって用いられる時間の単位で表わされるように適切な変換が行なわれる限り、インターバルは他の時間の単位で表わされも良い。更に、他の実施形態において、現在時刻はＵＴＣ以外の標準時に基づく。そのような実施形態において、時間の標準は、ウェブ・サーバ１０６およびストリーミング・メディア・サーバ１０４に固有のものである。また、エンド・ユーザ・プロセッサ１０２に時間を計算させ、かつ認証チケットを生成する際に用いられるべく、ウェブ・サーバ１０６へその値を送信させることもまた、本発明の範囲内であることが理解されよう。更に他の実施形態において、セキュリティ・インターバルは、標準時が所望の桁数まで単に切り下げられるように選択される。

【００３３】

一旦、ウェブ・サーバ１０６がハッシュに対して入力された値（アルゴリズム公

開暗号キー情報、プライベート・キー情報および時刻）を有すれば、ウェブ・サーバ１０６は、ハッシュ・アルゴリズムへの入力ストリングを生成する。（ステップ４１４）。本実施形態において、ハッシュ・アルゴリズムは「ＭＤ５」のメッセージ・ダイジェスト・アルゴリズムである。さらに、本実施形態においては、メディア・サーバ１０４およびウェブ・サーバ１０６は同じアルゴリズムを利用する。

【００３４】

チケットを生成する任意のハッシュあるいは暗号アルゴリズムを利用する事は、本質的に本発明の範囲内である事が理解されよう。更に、チケットを生成する２つのサーバ（先の実施形態においては、ウェブ・サーバ１０６およびストリーミング・メディア・サーバ１０４）は、同一の入力に基づいて同一のチケットを生成するか、同一の入力に基づいて、相互の偏差が所定の範囲内のチケットを生成する事が望ましい。他の実施形態において、セキュリティを増加させるために、複数の利用可能なアルゴリズムの１つが使用される。例えば、そのような実施形態は、複数の利用可能なアルゴリズムから任意に選択された１個のアルゴリズムを使用するか、もしくは、リクエストされたコンテンツ、リクエストの日付と時間、特定のエンド・ユーザ、コンテンツを有するエンティティ等に基づいて複数のアルゴリズムから１つを選択する事も可能である。このような実施形態において、システムは、ウェブ・サーバによって使用されるアルゴリズムの指示をメディア・サーバへ転送するか、あるいは、メディア・サーバは、ウェブ・サーバによって利用されるアルゴリズムと同一のアルゴリズムを選択・使用させるロジックを有する。

【００３５】

使用される特定のハッシュ・アルゴリズムに対して入力ストリングが有効である限り、およびストリーミング・メディア・サーバ１０４が入力ストリングの配置を認識している限り、入力値の任意の配置が入力ストリングとして使用されてよい。本実施形態において、次の所定の配置が使用される。

【００３６】

ＴＴＴＴＴＴＴＴＴＴＫＫＫＫＫＫＫＫＫＫＳＳＳＳＳＳＳＳＳＳＵＵＵＵＵＵＵＵ

・キーとセキュリティ・インターバル）を検索する。（ステップ５０６）。メディア・サーバ１０４はローカルメモリにプライベート・キー情報を格納するのが望ましいが、他の実施形態においては、メディア・サーバ１０４は、例えばマイクロソフト社によって提供されるＬＤＡＰによってアクセスされるアクティブ・ディレクトリー・ツリーに情報を格納してもよいし、または遠隔のデータベースに情報を格納してもよい。更に他の実施形態において、メディア・サーバ１０４は、ローカル・エリア・ネットワーク（ＬＡＮ）のようなネットワーク接続によってデータベース１０８にアクセスすることにより、プライベート・キー情報を検索する。

【００４１】

ウェブ・サーバ１０６が行ったように、メディア・サーバ１０４は更に現在時刻を計算し、セキュリティ・インターバルに最も近接した倍数にその値の端数を切り下げる。（ステップ５１０）。しかしながら、ウェブ・サーバ１０６と異なり、ストリーミング・メディア・サーバ１０４はメディア・サーバ１０４によって計算されたセキュリティ・インターバルに最も近接した倍数よりも過去の時間である、セキュリティ・インターバルに２番目に近接した倍数に現在時刻を切り下げた第２時刻を更に計算する（ステップ５１０）。更に、メディア・サーバ１０４はセキュリティ・インターバルに最も近接した倍数に現在時刻を切り上げた（つまり、時間的に後となる）第３時刻を計算する。（ステップ５１０）。

【００４２】

その後、メディア・サーバ１０４は、３つの対応するハッシュ入力ストリングを生成するために、検索されたプライベート・キー情報、受信された公開暗号キー情報および３つの時刻を使用する。（ステップ５１４）。その後、メディア・サーバ１０４は、ハッシュ・アルゴリズムに対して３つの入力ストリングを適用し、これによって、３枚のチケットが生成される。（ステップ５１８）。

【００４３】

個々にチケットを生成した後、メディア・サーバ１０４は、メディア・サーバ１０４によって生成されたチケットのうちの任意のチケットがウェブ・サーバ１０６によって生成されたチケットと一致するかどうかを判断する。（ステップ５２

ＵＵＵＵ。

【００３７】

ここで、「Ｔ」は時刻のデジットを表わし、「Ｋ」はセキュリティ・キーの英数字の文字を表し、「Ｓ」はストリーミングＩＤのデジットを表わし（任意の必要な先行・埋め込み文字を含む）、および「Ｕ」は、エンド・ユーザＩＤの英数字の文字を表わす（任意の必要な先行・埋め込み文字を含む）。他の実施形態において、入力ストリングは異なる長さであってもよい。

【００３８】

ハッシュ・アルゴリズム入力ストリングを生成した後、ウェブ・サーバ１０６は入力ストリングにハッシュ・アルゴリズムを適用し、それによってチケットを生成する。（ステップ４１８）。

【００３９】

ストリーミング・メディア・サーバ１０４が、リクエストされたコンテンツ・ストリーミングへのアクセスを許可するかどうかを判断するプロセスが、図５に関連してここに記載される。先ず、必ずしも必要と言うわけではないが、本実施形態におけるメディア・サーバ１０４が３枚の認証チケットを生成し、それぞれのチケットは異なる時刻に基づき、アクセスを許可するべきかどうかを判断するために使用される、と言う事に注目されるべきである。更に、ウェブ・サーバの機能性と同様に、アクセスを許可するべきかどうかを判断するプロセスは、メディア・サーバ１０４上に存在する許可ソフトウエア・コンポーネント中で実行されることが望ましい。

【００４０】

アクセスを許可するべきかどうかを判断する際に、ストリーミング・メディア・サーバ１０４は、エンド・ユーザのプロセッサ１０２上に存在するメディア・プレーヤーから、ストリーミングＩＤ、エンド・ユーザＩＤおよびチケットを含むストリーミング・リクエストを最初に受け取る。（ステップ５０２）。一旦ストリーミング・リクエストが受信されれば、メディア・サーバ１０４はハッシュ・アルゴリズムへの入力ストリングを生成する。この点において、メディア・サーバ１０４はローカルメモリからプライベート・キー情報（すなわちセキュリティ

２）。チケットが一致しない場合、ストリーミング・リクエストが本物でない、かつ（または）、有効期限切れである（つまり、メディア・サーバ１０４によって生成された時刻が、ユーザのリクエストの時間から測定されるセキュリティ・インターバルの範囲を超えている）可能性が高い。従って、メディア・サーバ１０４はリクエストされたコンテンツへのアクセスを許可しない。（ステップ５２６）。

【００４４】

チケットが一致する場合、ストリーミング・リクエストは本物かつセキュリティ・インターバルの範囲内のものである。しかしながら、アクセスを許可するのに先立って、メディア・サーバ１０４は、エンド・ユーザが既に同じコンテンツへのアクセスを以前にリクエストし、視聴したかどうかを最初に判断する。（ステップ５３０）。メディア・サーバ１０４は、エンド・ユーザＩＤのリストと、ユーザが以前にアクセスを許可されたストリーミングＩＤを対応させてローカルメモリに保持する事が望ましい。エンド・ユーザがリクエストされたコンテンツを既に視聴したかどうかを判断するために、メディア・サーバ１０４は、受信したエンド・ユーザＩＤおよびストリーミングＩＤが既に格納されているかどうかを判断するために、メモリにアクセスする。エンド・ユーザＩＤおよびストリーミングＩＤが既に格納されている場合、エンド・ユーザは、リクエストしたコンテンツに対するアクセスを許可されない。（ステップ５３０）。

【００４５】

受信されるエンド・ユーザＩＤおよびストリーミングＩＤが既に格納されていない場合、メディア・サーバ１０４は、エンド・ユーザＩＤおよびストリーミングＩＤをメモリに格納するステップへと進み（ステップ５３４）、エンド・ユーザにコンテンツへのアクセスを許可する。（ステップ５３８）。故に、エンド・ユーザＩＤおよびストリーミングＩＤを格納することによって、エンド・ユーザがリクエストしたコンテンツを指向するリンクを他人と共有する事を防ぐ事により、更に高度なセキュリティ保護が達成される。

【００４６】

ウェブ・サーバ１０６のローカル時間とメディア・サーバ１０４のローカル時間

の間が同期されていない事を補償するために、３つのチケットを使用する事が望ましい事が理解されよう。更に、ある状況では、エンド・ユーザが認証されている場合でも、メディア・サーバ１０４によって生成された第１のチケット（つまり、セキュリティ・インターバルに最も近接した倍数に切り下げた現在時刻に基づく）が、ウェブ・サーバ１０６によって生成された第１のチケットと一致しないであろう。例えば、所定のセキュリティ・インターバルを１５分とすると、ウェブ・サーバ１０６が午後１２：１４：００にチケットを生成し、メディア・サーバ１０４が午後１２：１６：００にその第１のチケットを生成する場合、同じ時間帯の同じ日において、たとえリクエストがセキュリティ・インターバルの範囲内にあってもチケットは一致しない。メディア・サーバ１０４が午後１２：１５：００に対応する時刻に基づいたチケットを生成する一方、ウェブ・サーバは、午後１２：００：００に対応する時刻に基づいたチケットを生成する。従って、本実施形態において、メディア・サーバ１０４は、セキュリティ・インターバルに２番目に近接した倍数に現在時刻を切り下げた時刻に基づいた第２のチケットを生成する。本例においては、午後１２：００：００に対応する。そのため、第２のチケットはウェブ・サーバ１０６によって生成されるチケットと一致する。同様に、セキュリティ・インターバルが経過した後、エンド・ユーザに対してアクセスを許可する事も可能である。したがって、本実施形態において、セキュリティ・インターバルは複数のチケットの使用を補償するように選択されるべきである。ウェブ・サーバ１０６およびメディア・サーバ１０４は、セキュリティ・インターバルの半分の時間内の誤差で合わせられた時計を有する事が望ましい。

【００４７】

メディア・サーバ１０４が、先の実施形態中における３つのチケットの代替として１つ以上の異なるチケットを生成する事もまた、本発明の範囲内である事が理解されよう。更に、先の実施形態においては、並列に生成されるチケットについて記述したが、メディア・サーバ１０４がチケットを直列に順次生成・比較する事もまた、本発明の範囲内である。更に、時刻は多くの方法で生成され得、例えば、メディア・サーバ１０４によって計算された第１の時刻から単にセキュリティ・インターバルを加える、または除する事によって生成されても良い、と言う事が理解されよう。

【００４８】

他の実施形態において、異なったレベルのセキュリティが提供されてもよい。特に、ウェブ・サーバ１０６によって生成されたチケットがメディア・サーバ１０４によって生成されたチケットのうちの１つと一致する場合、メディア・サーバ１０４は同じチケットが既に生成されていたかどうかを判断するステップへと進む。メディア・サーバ１０４は、アクセスが許可されたチケットのリストを保持する。論理上、このようなリストは全ての「使用済」チケットを表している。一致したチケットが「使用済」チケットのリストに載っていない場合、メディア・サーバ１０４は、エンド・ユーザのプロセッサ１０２に存在するメディア・プレーヤーに、リクエストしたコンテンツに対するアクセスを許可する。アクセスを許可するステップの一部として、メディア・サーバ１０４は、「使用済」チケットのリストを更に更新する。一致したチケットが使用済チケットのリストに載っている場合、メディア・サーバ１０４はアクセスを拒否し、リクエストしたエンド・ユーザに適切なメッセージを供給する。使用済チケットをトラッキングする事によって、システムは、認可されたエンド・ユーザがウェブ・サーバ１０６から受信されたストリーミング・リクエストを他人と共有するのを防ぐ。

【００４９】

さらに、アクセスを許可するべきかどうかを判断する際にエラー計算を使用する事も、本発明の範囲内であると言う事が理解されよう。例えば、１つのエラー計算は、所定の期間（例えば１５分、３０分等）、適用可能なセキュリティ・インターバルの設定百分率（例えば、５０％、１２５％等）のセット割合あるいは他のエラー計算のような、エラー・インターバルを現在時刻に加えた、および（または）除した値に基づいて１以上の追加のチケットを生成するメディア・サーバ１０４を伴う。このようなエラー計算は、先の実施形態において、第２時刻または第３時刻の代替値として、またはそれに加えて使用されてもよい。

【００５０】

他の実施形態において、ウェブ・サーバ１０６およびメディア・サーバ１０４は、時刻を先の実施形態とは異なった方法で計算することにより、チケットを生成する。典型的な一実施形態において、ウェブ・サーバ１０６およびメディア・サーバ１０４は現在時刻およびセキュリティ・インターバル以外の或るインターバルの倍数によって切り下げまたは切り上げた現在時刻を計算する。セキュリティ・インターバルが１５分である実施形態の場合、ウェブ・サーバ１０６は、５分のインターバルに最も近接した値に切り下げた現在時刻に基づいてチケットを生成する。次に、ストリーミング・メディア・サーバ１０４は、同じ５分インターバルに切り下げた現在時刻に基づいてチケットを生成する。チケットが一致しない場合、メディア・サーバ１０４は、時間的に遡った次のインターバルに切り下げた時間に基づいてチケットを生成するステップに進む。メディア・サーバは、所定の回数、またはウェブ・サーバとメディア・サーバのチケットが一致するまで、時間的に遡った次のよりインターバルに基づいたチケットを生成し続ける。メディア・サーバ１０４は、その和が少なくともセキュリティ・インターバルに及ぶ複数のタイム・インターバルに基づいて新しいチケットを繰り返し生成する。本実施例において、メディア・サーバ１０４は、合計１５分の５分インターバル、つまり、少なくとも３枚のチケットを生成する。

【００５１】

許可プロセスにおいてエンド・ユーザＩＤの使用を完全に省略するか、あるいは上述された方法とは異なる方法でエンド・ユーザＩＤを使用する事もまた、本発明の範囲内である事が理解されよう。例えば、他の実施形態において、エンド・ユーザＩＤは、ハッシュ・アルゴリズムへの入力ストリングの一部として使用されない。その代わり、データベース１０８は、どのエンド・ユーザがコンテンツへのアクセスをリクエストしたかをトラッキングするためのテーブルを含む。このような実施形態は、ストリーミングＩＤで識別されるコンテンツと、ユーザＩＤで識別され、該コンテンツストリーミングに既にアクセスまたは視聴したエンド・ユーザとを対応付けるレコードを含む視聴ユーザ（ストリーミング）テーブルを含む。同様に、本実施形態は、再生リストＩＤで識別される再生リストと、ユーザＩＤで識別され、該コンテンツストリーミングに既にアクセスしたか視聴したエンド・ユーザとを対応付けるレコードを含む視聴ユーザ（再生リスト）テーブルを含む。認証チケットを生成する前に、ウェブ・サーバは、同じエンド・ユーザが特定のストリーミングおよび再生リストにアクセスする事を既にリクエストしたかどうかを判断するために、適切な視聴ユーザ・テーブルをチェックする。エンド・ユーザが既にアクセスをリクエストしていた場合、ウェブ・サーバはアクセスを拒否するか、あるいは、今回のアクセスに対してエンド・ユーザは再び課金されるであろうということを示すウェブ・ページをエンド・ユーザに供給する。テーブルは、セキュリティ・インターバルのような期間あるいはそれを超過する期間の後に、自動的にクリアーされる。

【００５２】

本発明が、コンテンツの所有者であるクライアントに代わって、ウェブ・サーバ、ストリーミング・メディア・サーバおよび再生リスト・サーバを例えばサービス・プロバイダーが操作するような比較的複雑なシステムに適用される事も可能である、と言う事が理解されよう。このような実施形態が、図６～図８に関してここに記述される。本実施形態における機能性の多くは図３に示された実施形態における機能性と同一であり、同一の技術のうちで任意のものによって実施され得る、と言う事が当業者によって理解されよう。

【００５３】

図６に示されるように、本システムは図１に示された実施形態の構成要素と同様の構成要素を幾つか含む。本システムは、エンド・ユーザ・プロセッサ６０２、データベース６０８を含む１つ以上のストリーミング・メディア・サーバ６０４、および１つ以上のウェブ・サーバ６０６を含み、これらのすべてはインターネットあるいは他のネットワークへ接続される。さらに、本実施形態におけるシステムは、サービス・プロバイダーによって操作される再生リスト・サーバ６１０を更に含む。データベース６０８を含むウェブ・サーバ６０６、ストリーミング・メディア・サーバ６０４、および再生リスト・サーバ６１０は、ローカル・エリア・ネットワーク（ＬＡＮ）またはワイド・エリア・ネットワーク（ＷＡＮ）のようなサービス・プロバイダーのネットワークおよびインターネットに接続される事が好ましい。

【００５４】

一般的に、データベース６０８は図２の実施形態のデータベースに含まれている情報と同じ情報を含んでいるが、該情報はクライアント・アカウント毎に格納される。図７に示されるように、データベース６０８は、アカウントＩＤによって識別される各クライアントのレコードを含むアカウント・テーブル７０２を含む。各レコードは、クライアント名、アドレス、請求書発行情報等のクライアントを識別する情報（「クライアント情報」）、クライアントのコンテンツが保護されているかどうかに関する表示（「コンテンツ保護」）、クライアントのセキュリティ・キー（「セキュリティ・キー」）、およびセキュリティ・インターバル（「セキュリティ・インターバル」）を更に含む。

【００５５】

図２の実施形態と同様に、本データベース６０８は、ストリーミングＩＤによって識別される各コンテンツ・ファイル用のストリーミング識別情報を含むストリーミング・テーブル７０４と、許可されたユーザ・グループＩＤにエンド・ユーザＩＤを対応付ける許可ユーザ・テーブル７０６と、再生リストＩＤによって識別される各再生リスト用の再生リスト識別情報を含む再生リストテーブル７０８と、所定の再生リストＩＤに関連付けられたストリーミングＩＤを識別する再生リスト・ストリーミング・テーブル７１０を更に備える。図２のデータベースに関して記述された情報フィールドに加えて、ストリーミング・テーブル７０４および再生リストテーブル７０８は、各コンテンツ・ファイルおよび各再生リストに関連したアカウントＩＤを識別するフィールドをそれぞれに更に含む。

【００５６】

本データベース６０８は、コンテンツ・ファイルが格納される特定のストリーミング・メディア・サーバ６０４のホスト名を識別し、ストリーミングＩＤによって指定される各コンテンツ・ファイル用のレコードを含むストリーミング・サーバ・テーブル７１２を含む。図２におけるの実施形態と同様、ホスト名はメディア・サーバ６０４のＤＮＳ名である。

【００５７】

本実施形態の動作が、図８のワークフローに関してここに記述される。本実施例の例のために、エンド・ユーザは１つの保護コンテンツを含む再生リストへのアクセスをリクエストする。最初、ウェブ・サーバ６０６は、認証アプリケーションにログインすることをエンド・ユーザにリクエストし、所定のストリーミング・メディアを視聴する選択肢をエンド・ユーザに選択させるウェブ・ページを提供する。（ステップ８０２）。図３の実施形態と同様に、典型的なウェブ・ページは、リンクを起動する事によって特定のコンテンツ・ファイルを選択する事と、エンド・ユーザＩＤを提供する事と、請求書発行情報を提供する事をエンド・ユーザにリクエストするフォームを備え得る。ウェブ・ページに応答して、エンド・ユーザはエンド・ユーザのユーザＩＤおよびクレジットカード情報を提供し、ストリーミング・リクエスト・リンクを起動し、これによって、特定のストリーミング・メディア・コンテンツ・ファイルへのアクセスをリクエストする。再生リストＩＤが「７８９０００」である典型的なストリーミング・リクエスト・リンクは以下のとおりである。＜Ａ　ｈｒｅｆ”ｈｔｔｐ：／／ｐｌａｙｌｉｓｔｓｅｒｖｅｒ．ｃｏｍｐａｎｙ．ｃｏｍ／ｍａｋｅｐｌａｙｌｉｓｔ．ｄｌｌ？ＩＤ＝７８９０００”＞。

【００５８】

エンド・ユーザがストリーミング・リクエスト・リンクを起動する場合、エンド・ユーザ・プロセッサ６０２上で実行されるプログラミング・スクリプトによって、ストリーミング・リクエスト・リンクおよびエンド・ユーザＩＤがウェブ・サーバ６０６に送信される。（ステップ８０４）。当業者は、エンド・ユーザ・スクリプトは、例えばＣ＋＋、Ｐｅｒｌ、ビジュアルベーシック、Ｊａｖａ（登録商標）等の実質的に任意のプログラム言語によって記述され得る事を認識するであろう。本実施形態において、スクリプトはＪａｖａスクリプトであり、エンド・ユーザのウェブ・ブラウザと共働して実行される。

【００５９】

一旦ウェブ・サーバ６０６がスクリプトからストリーミング・リクエストを受け取れば、ウェブ・サーバ６０６は、許可ソフトウェア・プラグインの指示の下にチケットを生成する。この点において、ウェブ・サーバ６０６は、認証チケットを生成する際に使用されるプライベート・キーをデータベース６０８へ要求するリクエストを発行する（本実施形態において、セキュリティ・キーとセキュリティ・インターバルはリクエストされた再生リストと関連付けられている）。（ステップ８０６）。これを受けて、データベース６０８はウェブ・サーバ６０６にプライベート・キーを返す。（ステップ８０８）。

【００６０】

データベース６０８からプライベート・キーを取得した後、ウェブ・サーバ６０６は、図４に関して上に記述されるようなストリーミングされたものの代わりに再生リストＩＤを使用することによってチケットを生成する（本実施形態において、再生リストＩＤに代替される）。（ステップ８１０）。前述されたように、ウェブ・サーバ６０６は、チケットを生成するためにハッシュ・アルゴリズムにプライベート・キー、ストリーミングＩＤ、エンド・ユーザＩＤおよび時刻を適用する。その後、ウェブ・サーバ６０６は、エンド・ユーザ・プロセッサ６０２上で実行されるウェブ・ブラウザにチケットおよびエンド・ユーザＩＤを返す。（ステップ８１２）。

【００６１】

チケットを受信した後、エンド・ユーザ・プロセッサ６０２上で実行されるスクリプトは、ストリーミング・リクエスト・リンクの末端に情報をアペンドする。（ステップ８１４）。典型的なリンクが以下に示される。＜Ａ　ｈｒｅｆ　”ｈｔｔｐ：／／ｐｌａｙｌｉｓｔｓｅｒｖｅｒ．ｃｏｍｐａｎｙ．ｃｏｍ／ｍａｋｅｐｌａｙｌｉｓｔ？ＩＤ＝７８９０００＆ＴＩＣＫＥＴ＝ｕｖｗ１２３ｘｙｚ　ｒｅｔｕｎ＆ｕｓｅｒ＿ｉｄ＝ａｂｃ１２３ｄｅｆ”＞。ここで、再生リストＩＤは「７８９０００」で表され、チケットは「ｕｖｗ１２３ｘｙｚ」で表され、エンド・ユーザＩＤは「ａｂｃ１２３ｄｅｆ」で表される。

【００６２】

エンド・ユーザ・プロセッサ６０２上で実行されるスクリプトは、ホスト名「ｐｌａｙｌｉｓｔｓｅｒｖｅｒ．ｃｏｍｐａｎｙ．ｃｏｍ」によってストリーミング・リクエスト・リンク中に識別される再生リスト・サーバ６１０を呼び出させる。（ステップ８１６）。従って、再生リスト・サーバ６１０は、リンク、再生リストＩＤ、チケットおよびユーザＩＤを供給される。「ｍａｋｅｐｌａｙｌｉｓｔ．ｄｌｌ」オブジェクトの管理の下で、再生リスト・サーバ６１０は、ＡＳＸファイルのような、コンテンツがウインドウズ・メディア・フォーマットであるリダイレクター・ファイルを生成する。（ステップ８１８）。「ｍａｋｅｐｌａｙｌｉｓｔ」プログラムは、例えばＡＳＰを含む多くのプログラムあるいは技術のうちの任意のものを使用して実施されうる。リダイレクター・ファイルは、チケットおよび公開暗号キー（つまり、ストリーミングＩＤおよびエンド・ユーザＩＤ）に加えて、リクエストされたコンテンツへのリンクを含んでいる。リダイレクター・ファイルを生成するために、再生リストと、ストリーミングＩＤに関連付けられたホスト名・ＵＲＬ接頭部・ファイル名を含むコンテンツ・ファイルへのリンクに際して必要な情報とを含むコンテンツ・ファイルのストリーミングＩＤを検索するために、再生リスト・サーバ６１０はデータベース６０８にアクセスする。

【００６３】

他の実施形態において、エンド・ユーザ・スクリプトはストリーミング・リクエストにチケットをアペンドするために利用されない。その代わり、エンド・ユーザが自身のエンド・ユーザＩＤを提供し、ストリーミング・リクエスト・リンクを起動する場合（ステップ８０４）、ウェブ・サーバ６０６上で実行される認証アプリケーションがチケットを生成し、ストリーミング・リクエスト・リンクにチケットおよびエンド・ユーザＩＤをアペンドし、リダイレクター・ファイルを作成するために再生リスト・サーバ６１０を直接呼び出す。さらにウェブ・サーバ６０６がエンド・ユーザ・プロセッサ６０２上のメディア・プレーヤーを識別する情報を再生リスト・サーバ６１０情報に転送するので、再生リスト・サーバ６１０は、メディア・プレーヤーへリダイレクター・ファイルを転送する（ステップ８１２，８１４および８１６が省略される）。このような実施形態は、図１０に関して後述される。

【００６４】

その後、再生リスト・サーバ６１０はエンド・ユーザ・プロセッサ６０２のメディア・プレーヤーへＡＳＸリダイレクター・ファイルを転送する。（ステップ８２０）。本実施例のために、ＡＳＸファイルは以下のとおりに記述される。

【表２】

```
<ASX>
 <ENTRY>
   <REF
   href="mms://mediaserver.company.com/stream1.asf?ID=123456&TICKET=
     uvw123xyz&USER_ID=abc123def">
 </ENTRY>
</ASX>
```

【００６５】

ここでＵＲＬ接頭部は、「ｍｍｓ：／／」で表され、適切なメディア・サーバ６０４のホスト名が「ｍｅｄｉａｓｅｒｖｅｒ．ｃｏｍｐａｎｙ．ｃｏｍ」で表され、ファイル名は「ｓｔｒｅａｍ１．ａｓｆ」で表され、リクエストされたコンテンツ・ストリーミングＩＤは「１２３４５６」で表わされ、チケットは「ｕｖｗ１２３ｘｙｚ」で表され、エンド・ユーザＩＤは「ａｂｃ１２３ｄｅｆ」で表される。

【００６６】

リダイレクター・ファイルは、コンテンツ・ファイル用メタデータや、広告のような保護されていないファイルのような他の情報を含んでいて良い。

【００６７】

ＡＳＸファイルを受信した後、エンド・ユーザ・プロセッサ６０２はストリーミング・メディア・コンテンツをリクエストするステップへと進む。より詳細には、メディア・プレーヤーは、ＡＳＸファイル中に識別される「ｍｅｄｉａｓｅｒｖｅｒ．ｃｏｍｐａｎｙ．ｃｏｍ」（つまり、ストリーミング・メディア・サーバ６０４）を呼び出す。（ステップ８２２）。呼び出された後、メディア・プレーヤーは、リクエストされた番組のストリーミングＩＤのコピー、ウェブ・サーバ６０６によって生成されたチケット、およびエンド・ユーザＩＤをストリーミング・メディア・サーバに６０４を供給する。

【００６８】

メディア・プレーヤーの呼び出しに応答して、ストリーミング・メディア・サーバ６０４は、エンド・ユーザがアクセスをリクエストしているコンテンツへのアクセスを認めるかどうかを判断するステップへと進む。（ステップ８２４）。ストリーミング・メディア・サーバ６０４は、個々に１以上の認証チケットを生成し、そのチケットをウェブ・サーバ６０６によって生成されたチケットと比較することにより、アクセスを許可するかどうかを判断する。認証チケットを生成するおよび比較するプロセスは、ストリーミングＩＤの代わりに再生リストＩＤを使用する点を除いて、図５に関して記述されるのと同じ方法で達成される。メディア・サーバ６０４によって生成されたチケットがウェブ・サーバ６０６によって生成されたチケットと一致する場合、メディア・サーバ６０４はエンド・ユーザがアクセスをリクエストしているコンテンツへのアクセスを許可する。（ステップ８２４）。

【００６９】

先の実施形態はセキュリティ・キーとセキュリティ・インターバルの両方を含むプライベート・キーを利用するが、プライベート・キーとしてそれ以上、またはそれ以下の情報を利用するのも、本発明の範囲内である事が理解されよう。例えば、他の実施形態においてはセキュリティ・キーが使用されない。また、他の実施形態においては、例えばクライアントのユーザ名およびパスワードを含む補足情報が、プライベート・キーに含まれる。同様に、ストリーミングＩＤおよびエンド・ユーザＩＤ以外の情報を含む公開暗号キーを利用する事も、本発明の範囲内である。例えばファイル・パス名の様な、情報を識別する他のコンテンツ・ファイル識別情報が使用されてもよい。さらに、ある実施形態においては、エンド・ユーザＩＤが公開暗号キー情報から省略されてもよい。更に他の実施形態において、公開暗号キー情報は、他のリクエスト・コンテンツ・ファイルのタイトルあるいはストリーミング詳細のような補足情報を含んでいる。

【００７０】

ウェブ・サーバおよびストリーミング・メディア・サーバによって提供されると記載された機能性が、それに関連する他のデバイス上において実施され得るということが理解されよう。例えば、本発明の一実施形態において、ストリーミング・メディア・サーバが、それに接続される関連するアプリケーションサーバを備え、アプリケーションサーバは、コンテンツに対するアクセスを拒否または許可するプロセスの全部または一部を実施する。同様に、それは、例えば認証チケットを生成する過程を含むウェブ・サーバの機能性のうちの一部または全部を提供するためにウェブ・サーバとアプリケーションサーバとを関連付ける事は、本発明の範囲内である。そのため、特定のサーバに対する言及は、言及されたサーバと接続される他の関連するサーバあるいはプロセッサを含む、という事を意図している。

【００７１】

さらに、認証チケットを正確な時間に生成する必要は無い、と言う事が理解されよう。例えば、ウェブ・サーバによって生成されるチケットは、エンド・ユーザがストリーミング・リクエスト・リンクを起動する時間に基づいても良いし、ウェブ・サーバがデータベースからプライベート・キー情報を得る時間に基づいても良いし、ストリーミング・リクエストの起動時間に近接した任意の時間に基づいても良い。同様に、メディア・サーバが許可を生成する時間は、例えば、メディア・プレーヤーからの呼出がなされた時でも良いし、プライベート・キー情報が検索された後でも良いし、ストリーミング・リクエストの起動時間に近接した任意の時間であっても良い。更に、メディア・サーバが複数のチケットを生成する際、チケットは異なる複数の時間に基づいて生成されても良いし、単一の時間に基づいて生成されても良い。従って、時間または現在時刻に対する参照は、ある一定の範囲の参照を意図するものであり、正確な時刻の参照を意図するものではない。

【００７２】

先の典型的な実施形態は、単一のコンテンツに対するアクセスを管理すると言うコンテキストで記載されたが、先の実施形態が複数の保護コンテンツ・ファイルを含む再生リストへのアクセスを管理するために用いられ得るという事を当業者は理解するであろう。再生リストへのアクセスを管理するための典型的な一実施形態が、図６～図８の実施形態に関して記述される。このような実施形態は、下記の変更を伴った上で、前述の記載に従って作動する。一般的に、ウェブ・サーバ６０６は、各ストリーミングのストリーミングＩＤに基づいて、再生リストに含まれる各コンテンツ・ストリーミングのチケットを生成する。

【００７３】

エンド・ユーザ・プロセッサ６０２のメディア・プレーヤーは、再生リストＩＤを含むストリーミング・リクエストを再生リストプロセッサ６１０へ転送する。次に、再生リストプロセッサ６１０はリダイレクター・ファイルを生成し、メディア・プレーヤーにリダイレクター・ファイルを返送する。本実施例において、オブジェクト「ｍａｋｅｐｌａｙｌｉｓｔ．ｄｌｌ」は、適切なリダイレクター・ファイルを構築するために再生リストＩＤ「７８９０００」を使用する。より詳細には、どのコンテンツがリクエストされた再生リストの一部を構成するのかを判断するため、および再生リスト中のコンテンツ・ファイルの順番を判断するために、再生リスト・サーバ６１０は、再生リストテーブル７０８および再生リスト・ストリーミング・テーブル７１０にアクセスする。コンテンツ・ファイルのファイル名はストリーミング・テーブル７０４から検索される。続いて、エンド・ユーザ・プロセッサ６０２上で実行されるスクリプトは、対応するコンテンツ・ストリーミングへリンクするＵＲＬにストリーミングＩＤ、チケットおよびエンド・ユーザＩＤをアペンドする。本実施形態において、ストリーミング・サーバ・テーブル７１２中から識別されるように、全てのコンテンツ・ストリーミングは同一のメディア・サーバ６０４に格納される。

【００７４】

対応するコンテンツ・ストリーミングのためのＵＲＬリンクにアペンドされたストリーミングＩＤ、チケットおよびエンド・ユーザＩＤを含むＡＳＸリダイレクター・ファイルの例を以下に示す。

【表３】

```
<ASX>
 <ENTRY>
   <REF href="mms://mediaserver.company.com/stream1.asf? ID=123456&TICKET=
     abc111xyz&USER_ID=abc123def">
   <REF href="mms://mediaserver.company.com/stream2.asf? ID=234567&TICKET=
     def222xyz&USER_ID=abc123def">
   <REF href="mms://mediaserver.company.com/stream3.asf? ID=345678&TICKET=
     ghi333xyz&USER_ID=abc123def">
 </ENTRY>
</ASX>
```

【００７５】

その後、メディア・プレーヤーは、ストリーミング・メディア・サーバ６０４を連続して呼び出し、各呼び出しに対する各ＵＲＬリンクがリダイレクター・ファイルに含まれている。より具体的には、メディア・プレーヤーは先ず、第１のコンテンツ・ストリーミング（本例において、ストリーミングＩＤ「１２３４５６」を有する）にアクセスするためにメディア・サーバ６０４を呼び出す。呼び出しに応答して、また、図５に関して概説したように、メディア・サーバ６０４は個々にチケットを生成し、コンテンツへのアクセスを許可するべきかどうかを判断する。アクセスが許可されない場合、エンド・ユーザにその旨が通知される。一方、メディア・サーバがエンド・ユーザに対して第１のコンテンツ・ストリーミングへのアクセスを許可する場合、メディア・プレーヤーは、再生リスト中の残りのコンテンツ・ストリーミング用のメディア・サーバ６０４を呼び出すステップへと進む。それぞれの呼び出しで、メディア・サーバ６０４はリクエストされたコンテンツへのアクセスを許可するか拒否するかのステップへと進む。

【００７６】

しかしながら、このような実施形態において、各々のコンテンツ・ストリーミングが、再生リスト中のストリーミングに先立って再生されるコンテンツ・ストリーミングの所要時間の合計を計上した長さの個々のセキュリティ・インターバルを有することが望ましい、と言う事が理解されるべきである。例えば、それぞれが５分の所要時間を有する（ストリーミング・テーブル７０４のストリーミング詳細フィールドに示される）３つのコンテンツ・ストリーミングを含む再生リスト中で、第２のストリーミングのセキュリティ・インターバルは、第１のストリーミングのセキュリティ・インターバルよりも５分間だけ長くても良く、第３のストリーミングのセキュリティ・インターバルは、第１のストリーミングのセキュリティ・インターバルよりも１０分間だけ長くても良い。再生リスト中の各ストリーミングの所要時間を計上する事によって、システムは、許可されたエンド・ユーザが、再生リスト中の第１のコンテンツ・ストリーミングへのアクセスを許可されるがチケットが有効期限切れになったという理由によって後続のコンテンツ・ストリーミングに対するアクセスが許可されない、という事を防止する。更に、セキュリティ・インターバルは、再生リストに含まれる広告のような任意の非保護コンテンツの所要時間を計上しても良い。

【００７７】

更に他の実施形態は、再生リストＩＤに基づいてチケットを生成することにより、複数の保護コンテンツ・ストリーミングを含む再生リストへのアクセスを管理する。このような実施形態は、下記の変更を伴った上で、図６～図８のシステムの記述に従って作動する。一般的に、一旦エンド・ユーザが認証アプリケーションにログインし、再生リストへのアクセスをリクエストすれば、ウェブ・サーバ６０６は、再生リストＩＤに基づいてチケットを生成し、エンド・ユーザ・プロセッサ６０２にチケットを返送する。これを受けて、エンド・ユーザ・プロセッサ６０２上で実行されるスクリプトは、ストリーミング・リクエスト・リンクにチケットおよびエンド・ユーザＩＤをアペンドする。以下、公開暗号キー情報がアペンドされたストリーミング・リクエスト・リンクの例を示す。＜Ａ　ｈｒｅｆ”ｈｔｔｐ：／／ｐｌａｙｌｉｓｔｓｅｒｖｅｒ．ｃｏｍｐａｎｙ．ｃｏｍ／ｍａｋｅｐｌａｙｌｉｓｔ．ｄｌｌ？ＰＬＡＹＬＩＳＴ＿ＩＤ＝７８９０００＆ＴＩＣＫＥＴ＝ｘｙｚ３２１ａｂｃ＆ＵＳＥＲ＿ＩＤ＝ａｂｃ１２３ｄｅｆ”＞。ここで、再生リストＩＤは「７８９０００」で表され、チケットは「ｘｙｚ３２１ａｂｃ」で表され、エンド・ユーザＩＤは「ａｂｃ１２３ｄｅｆ」で表される。

【００７８】

エンド・ユーザ・プロセッサ６０２は、「ｐｌａｙｌｉｓｔｓｅｒｖｅｒ．ｃｏｍｐａｎｙ．ｃｏｍ」と言う名称によって識別される再生リスト・サーバ６１０を呼び出す。次に、再生リスト・サーバ６１０は、リダイレクター・ファイルを生成するために再生リスト・サーバ６１０に存在する「ｍａｋｅｐｌａｙｌｉｓｔ．ｄｌｌ」オブジェクトを始動する。本実施形態において、コンテンツ・ストリーミングはすべて同一のメディア・サーバ６０４上に存在する。先の実施形態と異なり、「ｍａｋａｐｌａｙｌｉｓｔ．ｄｌｌ」オブジェクトは、リダイレクター・ファイル中の第１のＵＲＬの端部に、再生リスト中の後続する保護コンテンツ・ストリーミングのファイル名をアペンドし、それぞれの後続のＵＲＬリンクには、再生リストＩＤおよびチケットのみがアペンドされる。ＡＳＸリダイレクター・ファイルの例を以下に示す。

【表４】

```
<ASX>
 <ENTRY>
    <REF href="mms://mediaserver.company.com/stream1.asf?PLAYLIST_ID=789000&
    TICKET=xyz321abc&USER_ID=abc123def&STREAM=stream2.asf&
    STREAM=stream3.asf">
    <REF href="mms://mediaserver.company.com/stream2.asf?PLAYLIST_ID=789000&
       &TICKET=xyz321abc">
    <REF href="mms://mediaserver.company.com/stream3.asf?PLAYLIST_ID=789000&
       &TICKET=xyz321abc">
 </ENTRY>
</ASX>
```

【００７９】

ここで、再生リストは、「ｓｔｒｅａｍ１．ａｓｆ」、「ｓｔｒｅａｍ２．ａｓｆ」、「ｓｔｒｅａｍ３．ａｓｆ」、と名前の付けられた３つのウィンドウズ・メディア・フォーマット（ウィンドウズは登録商標）のコンテンツ・ファイルを含み、再生リストＩＤは「７８９０００」で表され、エンド・ユーザＩＤは「ａｂｃ１２３ｄｅｆ」で表され、チケットは「ｘｙｚ３２１ａｂｃ」で表される。

【００８０】

エンド・ユーザ・プロセッサ６０２のメディア・プレーヤーは、第１のコンテンツ・ファイルにアクセスするために、ｍｅｄｉａｓｅｒｖｅｒ．ｃｏｍｐａｎｙ．ｃｏｍ（即ち、ストリーミング・メディア・サーバ６０４のホスト名）を呼び出すステップへ進む。メディア・サーバ６０４は、図５に関して上述されたように、再生リストＩＤに基づいてチケットを生成し、アクセスを許可または拒否するステップへと進む。メディア・サーバ６０４がアクセスを許可し、メディア・プレーヤーに再生リスト中の第１のコンテンツ・ファイルを提供する場合、メディア・サーバ６０４は、再生リストＩＤおよび対応するチケットのためにローカルに格納されたテーブル中にレコードを作成し、リダイレクター・ファイルに含まれている再生リスト中の後続するコンテンツ・ストリーミングのファイル名をレコード中に格納する。

【００８１】

メディア・プレーヤーが第２のコンテンツ・ストリーミングへのアクセスを要求する場合、メディア・プレーヤーはメディア・サーバ６０４に再生リストＩＤおよびチケットを供給する。次に、メディア・サーバ６０４は、再生リストＩＤおよびチケットによって識別されたレコードを、テーブル中から検索する。レコードが存在する場合、メディア・サーバ６０４は第２のストリーミングをへのアクセスを許可し、特定のチケットを有するエンド・ユーザによって視聴された旨、ストリーミングにフラグを立てる。無許可のエンド・ユーザが同じＵＲＬリンクを使用して第２のストリーミングにアクセスすることを試みる場合、再生リストＩＤおよびチケットに関するレコードにおいて、上述したように第２のストリーミングが視聴された旨を示すフラグが立っているので、メディア・サーバ６０４はアクセスを拒否する。同様のプロセスが、再生リスト中の残りのコンテンツ・ストリーミングへのアクセスを許可するために利用される。当業者によって認識されるように、本実施形態は、第１のストリーミングへのアクセス許可と、後続のストリーミングとの間の時間遅延に起因して再生リスト中の後続のストリーミングへのアクセスが不適切に拒否されるような如何なる可能性をも回避する。

【００８２】

当業者は、本発明の方法およびシステムが多数の応用を有し、多数の方法において実施され得、従って、以上記載された典型的な実施形態および実施例によって制限され無いという事を認識するであろう。さらに、当業者によって理解されるように、本発明の範囲は、本願明細書に記載されているシステム構成部分に対する従来公知の、および将来開発される変更および修正を含む。

【図面の簡単な説明】

【００８３】

【図１】本発明の一実施形態におけるシステムを示す略図である。

【図２】本発明の一実施形態におけるデータベース示す略図である。

【図３】本発明の一実施形態におけるワークフローを示す図である。

【図４】本発明の一実施形態におけるチケットを生成するプロセスを示すフローチャートである。

【図５】本発明の一実施例における、ストリーミング・メディア・コンテンツのアイテムを供給するべきかどうかを判断するプロセスを示すフローチャートである。

【図６】本発明の他の実施形態におけるシステムを示す略図である。

【図７】本発明の他の実施形態におけるデータベースを示す略図である。

【図８】本発明の他の実施形態におけるワークフローを示す略図である。

【符号の説明】

【００８４】

１０２...エンド・ユーザ・プロセッサ

１０４...メディア・サーバ、

１０６...ウェブ・サーバ、

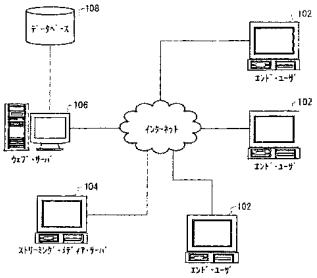
１０８...ＣＭデータベース

２０４...ストリーミング・テーブル

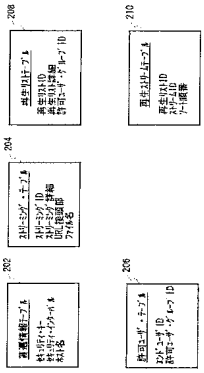
２０８...再生リストテーブル

２１０...再生リスト・ストリーミング・テーブル

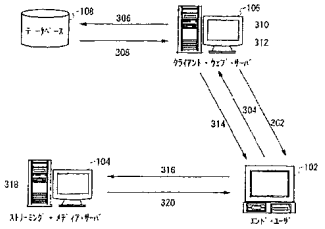
【図１】



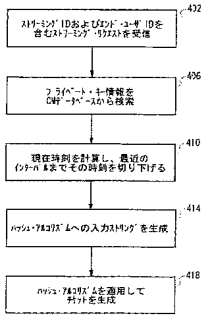
【図２】



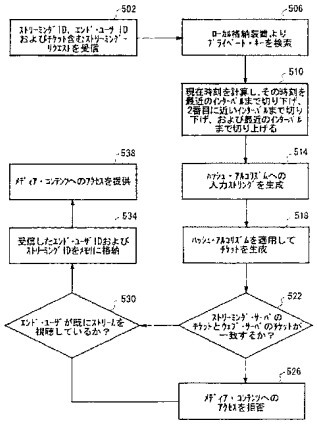
【図３】



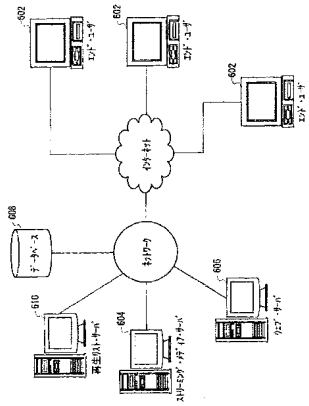
【図４】



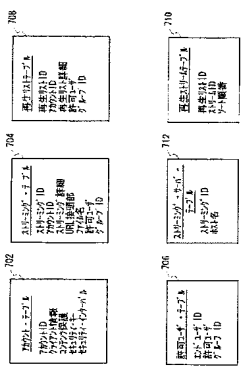
【図５】



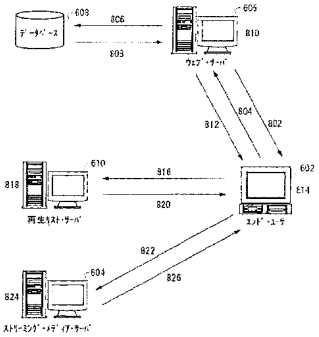
【図６】



【図７】



【図８】



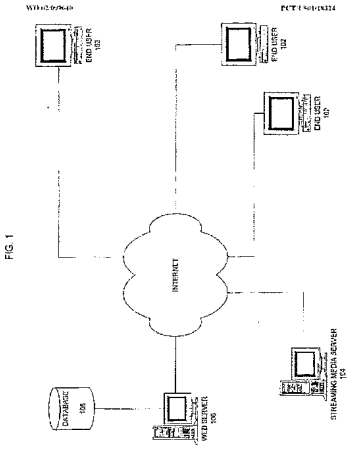


FIG. 1

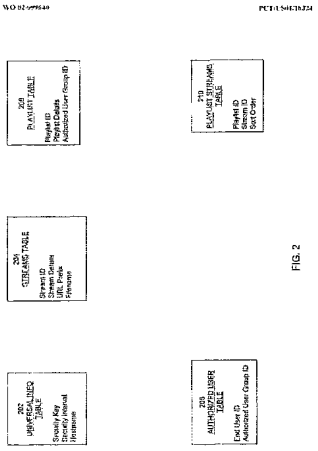


FIG. 2

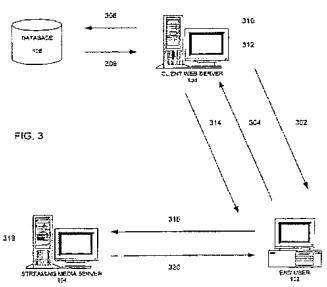


FIG. 3

FIG. 4

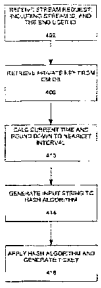


FIG. 5

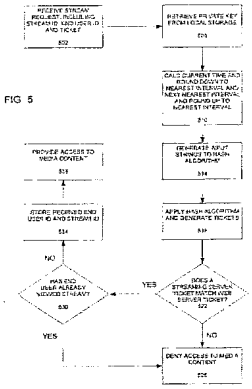


FIG. 6

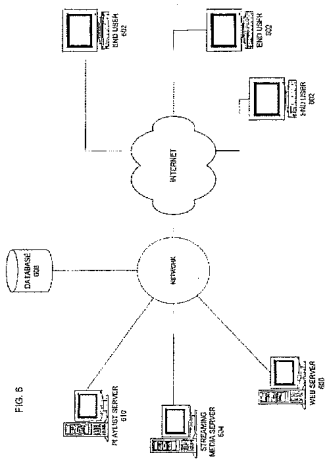


FIG. 7

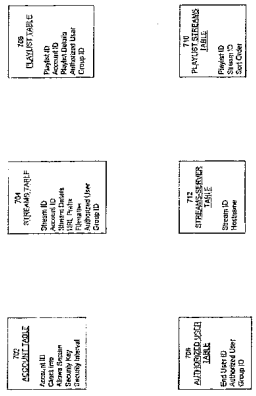
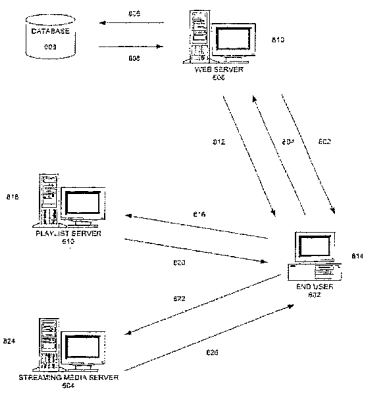


FIG. 8


DATABASE
WEB SERVER
PLAYLIST SERVER
END USER
STREAMING MEDIA SERVER

フロントページの続き

(72)発明者　チンタラ　アジャイ
アメリカ合衆国、７５００１　テキサス州、　アディソン、　２１０８　アディソン　ロード　アパートメント

Ｆターム(参考)　5B017 AA03　BA06　BB10　CA16
5B082 EA11　HA08

【国際調査報告】

INTERNATIONAL SEARCH REPORT

International application No.
PCT/US01/18324

A. CLASSIFICATION OF SUBJECT MATTER
IPC(7) : G06F 11/00, 1/24
US CL : 713/200, 201, 202, 100.
According to International Patent Classification (IPC) or to both national classification and IPC

B. FIELDS SEARCHED

Minimum documentation searched (classification system) followed by classification symbols)
U.S. : 713/200, 201, 202, 100.

Documentation searched other than minimum documentation to the extent that such documents are included in the fields searched

Electronic data base consulted during the international search (name of data base and, where practicable, search terms used)
East-DERWENT, JPO, EPO, USPAT FULL, PG PUB, IBMTDB. search terms: media, server, ticket, authorization, stream, data, pay, view, play, list.

C. DOCUMENTS CONSIDERED TO BE RELEVANT

Category*	Citation of document, with indication, where appropriate, of the relevant passages	Relevant to claim No.
Y	US 6,032,260 A (SASMAZEL et al.) 29 Feburary 2000 see entire document.	1-21
Y,E	US 6,263,432 B1 (SASMAZEL et al.) 17 July 2001, see entire document.	1-21

☐ Further documents are listed in the continuation of Box C. ☐ See patent family annex.

* Special categories of cited documents:
"A" document defining the general state of the art which is not considered to be of particular relevance
"E" earlier document published on or after the international filing date
"L" document which may throw doubts on priority claim(s) or which is cited to establish the publication date of another citation or other special reason (as specified)
"O" document referring to an oral disclosure, use, exhibition or other means
"P" document published prior to the international filing date but later than the priority date claimed
"T" later document published after the international filing date or priority date and not in conflict with the application but cited to understand the principle or theory underlying the invention
"X" document of particular relevance; the claimed invention cannot be considered novel or cannot be considered to involve an inventive step when the document is taken alone
"Y" document of particular relevance; the claimed invention cannot be considered to involve an inventive step when the document is combined with one or more other such documents, such combination being obvious to a person skilled in the art
"&" document member of the same patent family

Date of the actual completion of the international search	Date of mailing of the international search report
01 OCTOBER 2001	25 OCT 2001
Name and mailing address of the ISA/US Commissioner of Patents and Trademarks Box PCT Washington, D.C. 20231 Facsimile No. (703) 305-3230	Authorized officer NORMAN MICHAEL WRIGHT Telephone No. (703) 308-0000

Form PCT/ISA/210 (second sheet) (July 1998)*

【公報種別】特許法第１７条の２の規定による補正の掲載
【部門区分】第６部門第３区分
【発行日】平成17年12月22日(2005.12.22)

【公表番号】特表2004-533690(P2004-533690A)
【公表日】平成16年11月4日(2004.11.4)
【年通号数】公開・登録公報2004-043
【出願番号】特願2003-502687(P2003-502687)
【国際特許分類第７版】
Ｇ０６Ｆ　12/14
Ｇ０６Ｆ　12/00
【ＦＩ】
Ｇ０６Ｆ　12/14　５２０Ｄ
Ｇ０６Ｆ　12/14　５２０Ｆ
Ｇ０６Ｆ　12/00　５３７Ａ

【手続補正書】
【提出日】平成16年7月27日(2004.7.27)
【手続補正１】
【補正対象書類名】特許請求の範囲
【補正対象項目名】請求項１
【補正方法】変更
【補正の内容】
【請求項１】
１つ以上のメディア・ファイルへのアクセスを管理する方法であって、
ａ．第１時間に基づいて第１認証チケットを生成するステップと、
ｂ．ストリーミング・リクエストおよび前記第１認証チケットをサーバへ送信するステップと、
ｃ．第２時間に基づいて第２認証チケットを生成するステップと、
ｄ．前記メディア・ファイルへのアクセスを許可するかどうかの判断を可能にするために、前記第１認証チケットと前記第２認証チケットとを比較するステップと
を備える方法。
【手続補正２】
【補正対象書類名】特許請求の範囲
【補正対象項目名】請求項１２
【補正方法】変更
【補正の内容】
【請求項１２】
前記第２認証チケットを生成するステップがおおよそ前記第２時間に起こり、前記第２認証チケットは、前記第２時間を或る時間間隔の倍数まで切り下げた値と近似する第２時刻に更に基づき、
前記第３認証チケットを生成するステップがおおよそ前記第３時間に起こり、前記第３認証チケットは、前記第３時間を前記時間間隔の他の倍数まで切り上げた値と近似する第３時刻に更に基づく、請求項９に記載の方法。
【手続補正３】
【補正対象書類名】特許請求の範囲
【補正対象項目名】請求項３０
【補正方法】変更
【補正の内容】
【請求項３０】

１つ以上のメディア・ファイルへのアクセスを管理するシステムであって、
ソフトウエアと共働して、第１時間に基づいて第１認証チケットを生成する第１プロセッサと、
前記第１プロセッサからは独立しており、ソフトウエアと共働して、第２時間に基づいて第２認証チケットを生成する第２プロセッサと、
ソフトウエアと共働して、前記第１認証チケットを受信し、前記第１認証チケットと前記第２認証チケットとを比較する事によって、前記メディア・ファイルへのアクセスを許可するかどうかを判断する第３プロセッサと
を備えるシステム。
【手続補正４】
【補正対象書類名】明細書
【補正対象項目名】０００２
【補正方法】変更
【補正の内容】
【０００２】
インターネットとワールドワイド・ウェブの普及により、ストリーミング・メディア・コンテンツのようなデジタル・コンテンツの配信に関連する産業が発展した。例えば、ストリーミング・メディアは、エンターテインメント、通信教育および企業目的を含む多数の目的のうちの何れかのために使用され得る。エンターテインメント会社は映画とスポーツイベントをストリーミングし、通信教育会社は教育のコンテンツをストリーミングし、また、企業はトレーニング教材をストリーミングする。
【特許文献１】米国特許第６，０３２，２６０号明細書
【特許文献２】米国特許第６，２６３，４３２号明細書